



# Tronco: Um Operador Genético para Remapeamento de Serviços de Rede Sensível ao Histórico

Guilherme Werneck de Oliveira<sup>1,2</sup>, Elias P. Duarte Jr.<sup>1</sup>, Vinícius Fulber-Garcia<sup>1</sup>

<sup>1</sup>Universidade Federal do Paraná  
Departamento de Informática – Curitiba – PR – Brasil

<sup>2</sup>Instituto Federal do Paraná  
Pinhais – PR – Brasil

guilherme.oliveira@ifpr.edu.br, {elias,vinicius}@inf.ufpr.br

**Resumo.** *Uma das principais vantagens do paradigma da Virtualização de Funções de Rede (NFV) é a flexibilidade trazida para o núcleo da rede. Entretanto, é notável a lacuna de estratégias para o mapeamento de serviços de rede baseados em NFV que levam em conta que as redes de comunicações são dinâmicas, no sentido de que seu estado se altera gradualmente com o tempo. Por isso, este artigo apresenta um novo operador genético para o mapeamento dinâmico de serviços virtualizados em diferentes domínios e pontos de presença. Apesar de soluções de mapeamento baseadas em algoritmos genéticos apresentarem bons resultados para o processo de mapeamento inicial, elas desconsideram informações históricas para o remapeamento do serviço quando o estado da rede muda. O operador aqui proposto, denominado Tronco, utiliza dados históricos para garantir a otimalidade (local ou global) do remapeamento de serviços de rede. Os resultados experimentais de sua implementação são apresentados para diversos cenários, considerando: (i) o mesmo número de gerações; (ii) a execução até a convergência; e (iii) diferentes níveis de mudança no estado da rede. Os resultados demonstram melhorias significativas no remapeamento de serviços virtualizados com o uso do Tronco.*

## 1. Introdução

As redes de computadores são notavelmente dinâmicas, sujeitas a modificações frequentes e, tipicamente, graduais em seu estado, decorrentes de um conjunto complexo de interações entre sistemas, serviços e usuários [Oliveira et al. 2026]. Essa característica é denominada neste trabalho como **vivacidade da rede**. Todo um conjunto de paradigmas e tecnologias com avanço na última década visa enfrentar, entre outros, o problema da inflexibilidade do núcleo da rede, incluindo *Software-Defined Networking* (SDN) [Benzekki et al. 2016], *Network Function Virtualization* (NFV) [Rehman et al. 2019] e *Programmable Data Planes* (PDP) [Han et al. 2020]. Em particular, o paradigma NFV desacopla as funções de rede do hardware associado, implementando-as em um plano de software flexível, que permite a pronta execução de operações de ciclo de vida, como implantação, escalonamento e migração de funções e serviços de rede [Zhang et al. 2020].

Entre as operações do ciclo de vida, a implantação de funções e serviços NFV é um processo bastante complexo, que envolve o mapeamento de serviços em diferentes domínios e pontos de presença, a integração das funções à infraestrutura de

virtualização disponível e o agendamento da execução das mesmas nessa infraestrutura [Herrera and Botero 2016]. Ressalta-se que, de maneira geral, um serviço de rede consiste na composição de diversas funções encadeadas entre si. Diversas propostas para a otimização do mapeamento de serviços de rede vêm sendo apresentadas, com destaque para aquelas baseadas em algoritmos genéticos multiobjetivo, devido à sua notável adaptabilidade e efetividade em diversos cenários [Li et al. 2020, Fulber-Garcia et al. 2023, Fulber-Garcia et al. 2024a, Liu et al. 2024].

Porém, soluções de mapeamento baseadas em algoritmos genéticos são executadas considerando apenas um estado específico da rede, um *snapshot* da rede tirado no momento de sua execução. Ao considerar a vivacidade da rede, é possível perceber que frequentemente esses resultados do mapeamento estático degradam ao longo do tempo, de maneira que resultados antes otimizados podem se tornar subótimos para um novo estado atualizado da rede. Estratégias de monitoramento de funções e serviços, bem como de observabilidade de rede, podem detectar tais degradações e disparar novos processos de mapeamento, restabelecendo sua otimalidade (local ou global). A partir desse gatilho, as soluções iniciam sua execução considerando o estado mais atual da rede e determinam um novo resultado, sendo tal processo denominado remapeamento [Souza et al. 2020].

No entanto, soluções existentes de mapeamento baseadas em algoritmos genéticos multiobjetivo executam remapeamentos “a frio”, sem aproveitar resultados históricos de execuções anteriores para o mesmo serviço. Em situações normais, a vivacidade da rede preconiza que as mudanças de estado são graduais e, em alguns casos, até mesmo previsíveis. Ou seja, desde que nenhum evento catastrófico ocorra, causando uma interrupção na rede, partes de resultados anteriormente considerados ótimos para um estado passado podem permanecer válidas para o estado atual, desde que recombinadas estrategicamente.

Nesse contexto, este trabalho propõe um novo operador genético denominado **Tronco**, que permite manipular da população inicial usando resultados históricos, previamente observados na rede. O nome do operador é baseado no conceito homônimo da filogenética, em que um Tronco consiste em um conjunto de organismos extintos que apresentam parentesco próximo a um grupo de organismos modernos, mas que não possuem todas as suas características definidoras [Ward 2010]. O principal objetivo do novo operador genético é melhorar o remapeamento de um serviço de rede virtualizado em diferentes cenários de mudança do estado da rede. Para isso, consideram-se três hipóteses principais, segundo as quais o uso do Tronco, quando disponível, permite que a solução retorne melhores resultados em relação às estratégias tradicionais para: (i) o mesmo número de gerações; (ii) a execução até a convergência; e (iii) diferentes níveis de mudança do estado da rede.

O operador de Tronco foi implementado na solução *Genetic Service Mapping* (GeSeMa) [Fulber-Garcia et al. 2023], e avaliado sob diferentes cenários de mapeamento, mudanças de estado da rede e remapeamentos. Os resultados confirmaram as hipóteses levantadas, demonstrando que os retornos obtidos a partir do remapeamento utilizando o operador de Tronco foram superiores segundo a métrica de fronteiras relativas de Pareto. Esse resultado abre uma janela de oportunidades relacionadas ao aproveitamento de dados históricos em processos de otimização contínua da rede, tanto em NFV quanto em outros paradigmas de *softwarização* de redes, assim como em aplicações variadas em que algoritmos genéticos ou estratégias evolutivas sejam usados.

O restante deste artigo está organizado da seguinte forma: A Seção 2 introduz os principais conceitos de algoritmos genéticos utilizados ao longo do artigo; a Seção 3 apresenta os trabalhos relacionados; a Seção 4 detalha o operador de Tronco e discute sua aplicação no contexto do processo de remapeamento; a Seção 5 apresenta a integração do operador de Tronco na solução GeSeMa, bem como os testes conduzidos e os resultados obtidos; finalmente, a Seção 6 conclui o artigo.

## 2. Algoritmos Genéticos: Fundamentos e Aplicações

Algoritmos genéticos baseiam-se no princípio da evolução de Darwin e são aplicados à resolução de diferentes tipos de problemas de otimização [Forrest 1996, Lambora et al. 2019]. Um dos principais desafios relacionados ao uso de algoritmos genéticos está na modelagem do problema a ser resolvido, de forma que este possa ser compreendido à luz do processo de seleção natural das espécies. Para isso, o problema precisa ser modelado considerando, de maneira geral, a seguinte estrutura:

- **Indivíduo:** representação de um possível resultado capaz de solucionar completamente um dado problema; um indivíduo é composto por pelo menos um gene;
- **Gene:** representação de um subproblema que compõe o problema completo; a resolução do problema completo, por sua vez, depende da determinação, por meio de alelos, de todos os seus genes;
- **Alelo:** indicação de um resultado concreto para a solução de um subproblema; ou seja, um alelo é pontualmente atribuído a um gene.

O conjunto de indivíduos criado em um determinado momento do processo de otimização é chamado de **população**, e cada população retornada por uma iteração do algoritmo, *i.e.*, a população em um determinado momento no tempo, é chamada de **geração**. O conjunto de indivíduos que compõe a primeira geração, normalmente criado de forma aleatória, é chamado de **população inicial**, e é a partir dele que uma série de operadores genéticos são aplicados iterativamente em busca de ótimos, locais ou globais, mais adequados para a resolução do problema. Os principais operadores genéticos são:

- **Produção:** operador de gênese de indivíduos, sem que estes tenham, necessariamente, ancestrais. O operador de geração mais comum consiste na aleatorização de alelos para os genes do indivíduo, mas heurísticas de produção também podem ser utilizadas;
- **Seleção:** operador dedicado à escolha de indivíduos de uma geração para, por exemplo, passarem pelos processos de cruzamento e mutação, ou para serem encaminhados diretamente para uma geração posterior (**elitismo** dos melhores indivíduos). Os operadores não elitistas mais comuns são os de roleta, torneio e ranqueamento [Shukla et al. 2015];
- **Cruzamento:** operador capaz de criar um ou mais novos indivíduos para compor uma geração  $G^{m+1}$ , a partir do cruzamento de indivíduos de uma geração  $G^m$ . Alternativas bastante utilizadas consistem nas técnicas *Simulated Binary Crossover* (SBX) e *Half Uniform Crossover* (HUX) [Pachau et al. 2021];
- **Mutação:** operador vinculado à realização de modificações pontuais em alelos específicos associados aos genes de um indivíduo. Processos de troca de alelos entre genes diferentes (translocação) e de aleatorização de alelos (mutação natural) são amplamente utilizados [De Falco et al. 2002]. No entanto, podem ser adotadas heurísticas específicas orientadas ao problema que está sendo resolvido.

Esses operadores são estrategicamente acionados para promover o processo evolutivo, ajustando suas probabilidades de execução de modo a balancear a exploração (busca por regiões promissoras) e o aprofundamento (busca pelos melhores indivíduos nessas regiões) no espaço de busca associado ao problema. Todos os indivíduos gerados são avaliados conforme uma função objetivo, modelada a partir das métricas de interesse de otimização e de um objetivo, normalmente representado pela maximização, minimização ou estabilização dessas métricas.

Quando a otimização considera uma função mono-objetivo (um objetivo aplicado a uma única métrica), a comparação entre os indivíduos é direta e trivial. No entanto, quando a função é multiobjetivo (um ou mais objetivos aplicados a duas ou mais métricas), um processo distinto de comparação precisa ser estabelecido. Nesse caso, a técnica mais comum é a de **dominância de Pareto** [Voorneveld 2003]. Nessa técnica, estabelecem-se múltiplas fronteiras de dominância, nas quais os indivíduos presentes em uma fronteira  $F^n$  ( $n \geq 0$ ) não manifestam relação de dominância entre si; ou seja, cada indivíduo apresenta pelo menos uma métrica cuja avaliação é melhor em relação aos demais indivíduos da mesma fronteira. Por outro lado, existe pelo menos um indivíduo em uma fronteira  $F^n$  que apresenta avaliações melhores para todas as métricas em relação àqueles presentes em uma fronteira  $F^{n+x}$  ( $x > 0$ ). Assim, é possível determinar que a fronteira mais externa,  $n = 0$ , contém os indivíduos não dominados por nenhum outro, com os melhores resultados possíveis e sendo denominada **fronteira de Pareto**.

Cabe ressaltar, no entanto, que a dominância de Pareto pode ser calculada de maneira absoluta ou relativa [Fulber-Garcia et al. 2024b]. A dominância de Pareto absoluta é aquela calculada quando considerados todos os resultados possíveis para um dado problema. Já a **dominância de Pareto relativa** é aquela calculada a partir de um conjunto não exaustivo de resultados válidos para o problema. Esse cenário é comum quando se atinge o critério de parada da otimização antes de se exaurirem todas as possibilidades de resultados do problema. Nesse sentido, destaca-se também que os **critérios de parada** mais comuns adotados em otimizações baseadas em algoritmos genéticos são o número máximo de gerações, o tempo máximo de execução e a convergência. Particularmente, a convergência da otimização é normalmente determinada quando, após  $n$  gerações ou um tempo  $t$ , não há mudança na fronteira de Pareto relativa calculada entre gerações sucessivas ao longo do processo de otimização.

Algoritmos genéticos são especialmente adequados para problemas de otimização com grande espaço de busca, além de serem maleáveis por permitirem, sem mudanças no modelo de execução, a alteração da função objetivo avaliada. Naturalmente, essa classe de algoritmos não garante a obtenção de ótimos globais; entretanto, apresenta grande sucesso na obtenção de bons resultados, segundo um critério predefinido, em tempo de execução viável. Esse contexto se verifica em diversos problemas de otimização relacionados ao paradigma NFV, principalmente aqueles ligados à composição, integração, agendamento de execução, escalonamento e migração de serviços de rede virtualizados, tornando os algoritmos genéticos uma escolha apropriada para a resolução desses problemas [Rankothge et al. 2015].

### 3. Trabalhos Relacionados

O problema do mapeamento e remapeamento de serviços virtualizados de rede em múltiplos domínios e pontos de presença, resolvido como parte da implantação de um serviço, apresenta uma organização específica capaz de provocar a explosão do tamanho do espaço de busca, mesmo quando este pode ser restringido por políticas e limitações previamente definidas. Em geral, o tamanho do espaço de busca é dado por um arranjo com repetição  $AR_{n,p}$ , em que  $n$  representa a quantidade de domínios e pontos de presença disponíveis para o mapeamento, e  $p$  a quantidade total de funções de rede do serviço a serem mapeadas. Em razão disso, a literatura recente apresenta diversas soluções de mapeamento baseadas em algoritmos genéticos, sendo algumas das principais descritas ao longo desta seção.

Os autores em [Li et al. 2020] propõem uma solução implementada como um algoritmo genético mono-objetivo para a execução do mapeamento de serviços virtualizados de rede, visando à otimização de um indicador composto de qualidade de serviço, representado por  $E$ . Esse indicador incorpora, entre outros aspectos, medições ajustadas de métricas como a disponibilidade de enlaces interdomínios, a largura de banda interdomínios e a capacidade de hospedagem de funções de rede em cada domínio. A solução também permite ao usuário optar pelo mapeamento de funções de *backup* do serviço, visando ao aumento de sua confiabilidade.

A solução proposta em [Rodis and Papadimitriou 2021], por sua vez, também é baseada em um algoritmo genético mono-objetivo aplicado ao problema de mapeamento. Contudo, considera uma função objetivo voltada à minimização dos recursos residuais da infraestrutura provida pelos domínios e enlaces disponíveis. Para isso, todas as métricas relacionadas à disponibilidade de recursos computacionais e de comunicação são ajustadas e condensadas em um único indicador, denotado por  $C_m$ , o qual é diretamente avaliado no processo de otimização.

O *Genetic Service Mapping* (GeSeMa), apresentado em [Fulber-Garcia et al. 2023], consiste em uma solução de mapeamento multiobjetivo baseada em duas meta-heurísticas genéticas: SPEA2 e NSGA-II. A principal vantagem descrita para o GeSeMa é sua alta flexibilidade na personalização da função objetivo, a qual pode ser definida pelos usuários por meio de um documento YAML, sem exigir nenhuma modificação no código-fonte da solução. Em função disso, os autores destacam sua especial aplicabilidade no contexto de *marketplaces* NFV e de provedores de infraestrutura de virtualização, que precisam se adaptar de forma rápida e dinâmica às necessidades de seus clientes.

A solução denominada *Service Mapping Expedient for Networked Traffic and Environments* (SeMENTE) [Fulber-Garcia et al. 2024a] consiste em um algoritmo genético multiobjetivo baseado em NSGA-II, especialmente desenvolvido para o mapeamento de serviços virtualizados de rede implantados no modelo Multi-SFC [Huff et al. 2020], considerando suas particularidades arquiteturais e operacionais. O SeMENTE adota uma função objetivo mista, que visa à minimização do custo total de implantação e da latência interdomínios, ao mesmo tempo em que maximiza a largura de banda interdomínios.

Por fim, a solução proposta em [Liu et al. 2024], denominada GA-MOEA/D, fundamenta-se em um algoritmo genético multiobjetivo voltado ao mapeamento de

serviços virtualizados de rede *multicast*, considerando suas especificidades de roteamento de tráfego. Para isso, os autores baseiam a solução no *framework* MOEA/D e desenvolvem operadores genéticos próprios que, ao longo do processo de otimização, buscam minimizar a sobrecarga do mapeamento e o atraso total de transmissão de dados no serviço.

Apesar das diferentes soluções de mapeamento baseadas em algoritmos genéticos apresentadas, o remapeamento permanece um tema pouco explorado nesse contexto. Em particular, dentre as soluções destacadas nesta seção, apenas uma, o SeMENTE [Fulber-Garcia et al. 2024a], discute diretamente a questão do remapeamento de serviços, apresentando testes preliminares que se mostraram promissores quanto ao uso de resultados históricos para a execução desse processo. No entanto, o SeMENTE aborda exclusivamente o mapeamento de serviços no modelo Multi-SFC, além de não apresentar um mecanismo ou operador específico para a inserção dos resultados históricos durante a execução do remapeamento. Dessa forma, este trabalho se diferencia dos demais em três aspectos principais: (i) a formalização de um operador genético para a utilização de resultados históricos em processos de remapeamento; (ii) a implementação desse operador em uma solução flexível, capaz de realizar mapeamentos personalizados conforme a necessidade do usuário; e (iii) a discussão dos resultados experimentais relacionados ao uso desse operador de forma holística, considerando diversos casos de teste.

#### 4. O Operador de Tronco: Definição

Visando permitir que resultados históricos de mapeamentos baseados em algoritmos genéticos de um dado serviço, *i.e.*, resultados provenientes de processos anteriores de mapeamento, sejam utilizados para inicializar “a quente” um processo de remapeamento, propõe-se o operador de Tronco. Como mencionado anteriormente, no contexto da filogenética e da linhagem evolutiva, um Tronco consiste em um conjunto de organismos extintos que apresentam parentesco próximo a um grupo de organismos modernos, mas que não possuem todas as suas características definidoras [Ward 2010].

Ao aproximar esse conceito evolutivo do problema de otimização de mapeamento tratado neste artigo, considera-se que os resultados obtidos para estados anteriores da rede sejam classificados em um estado posterior e atualizado (que se modifica em função da vivacidade da rede) como extintos, uma vez que representam soluções ótimas que, potencialmente, já não existem. Entretanto, como interrupções na rede são, em geral, raras, tais resultados ótimos anteriores (organismos extintos) tendem a preservar características (partes dos resultados de mapeamento) semelhantes às dos ótimos atuais, configurando-os como parentes próximos que carregam informações relevantes para a identificação de novos resultados ótimos.

Considerando a descrição apresentada, estabelecem-se as seguintes notações:  $S$  representa um serviço de rede virtualizado, em suas diferentes possibilidades de composição de cadeia [Fulber-Garcia et al. 2020], que deve ser mapeado nos domínios e pontos de presença de uma rede de computadores;  $T$  é o conjunto  $\{t_1, t_2, \dots, t_{n-1}\}$ , que contém os instantes de tempo nos quais um processo de mapeamento ou remapeamento foi executado com sucesso para  $S$ ;  $E^{t_n}$  representa o estado da rede de computadores em um instante  $t_n$ ;  $R_{E^{t_n}}^S$  consiste no conjunto de resultados ótimos, locais ou globais, para o mapeamento do serviço  $S$ , considerando o estado da rede  $E^{t_n}$ . Sendo assim, o operador de Tronco aplicável ao remapeamento do serviço  $S$  no estado da rede  $E^{t_n}$  é denotado por

$Tronco_{E^{t_n}}^S$ , e o conjunto de resultados (indivíduos) que o compõem é definido conforme expresso na Equação 1.

$$Tronco_{E^{t_n}}^S \subseteq \bigcup_{i \in T} R_{E^{t_i}}^S \quad (1)$$

É importante ressaltar, entretanto, que a definição de Tronco apresentada é de caráter amplo, pois estabelece apenas sua condição necessária. Em função das características do algoritmo genético adotado e do processo de otimização considerado, podem ser definidos operadores de Tronco mais restritivos, desde que preservem a observância da Equação 1. Como exemplo, para reforçar a recência do Tronco e mitigar os efeitos de mudanças acumuladas decorrentes da vivacidade da rede ao longo de períodos extensos, é possível delimitar o Tronco de modo a considerar apenas os resultados históricos do último processo de mapeamento ou remapeamento executado. Nessa configuração, o operador de Tronco é então definido segundo a Equação 2.

$$Tronco_{E^{t_n}}^S \subseteq R_{E^{t_{n-1}}}^S \quad (2)$$

Outro aspecto relevante a ser considerado refere-se aos critérios de inclusão de resultados (indivíduos) no conjunto  $R_{E^{t_n}}^S$ . Em otimizações mono-objetivo, é possível, por exemplo, adotar uma postura elitista e incluir os  $n$  melhores resultados encontrados ao longo de todas as gerações ou, alternativamente, incluir a totalidade dos resultados obtidos na última geração (iteração) do algoritmo; ainda, pode-se optar pela inclusão apenas do melhor resultado encontrado. No caso de otimizações multiobjetivo, é possível adotar uma estratégia de inclusão apenas, por exemplo, da fronteira de Pareto relativa calculada na última geração executada pelo algoritmo, ou ainda das  $n$  melhores fronteiras. Todas essas decisões impactam diretamente a forma como o Tronco poderá atuar em remapeamentos futuros do serviço, sendo essas decisões dependentes de uma análise das características específicas do problema tratado e do algoritmo empregado.

O operador de Tronco deve ser executado considerando o ciclo de vida de uma otimização baseada em algoritmos genéticos, antes do operador de produção, e deve cumprir as seguintes tarefas: (i) validação dos indivíduos do Tronco, dada uma mudança no estado da rede (de  $E^{t_{n-1}}$  para  $E^{t_n}$ ): todos os indivíduos do Tronco devem ser verificados quanto à sua validade no novo estado (nesse momento, o indivíduo não é avaliado quanto à função objetivo, sendo apenas verificada sua viabilidade de existência no novo estado da rede); e (ii) inserção de uma quantidade  $k$  de indivíduos válidos do Tronco na população inicial, conforme uma estratégia predefinida pelo desenvolvedor da solução de mapeamento, sendo  $k$  definido pelo usuário e menor ou igual ao tamanho da população.

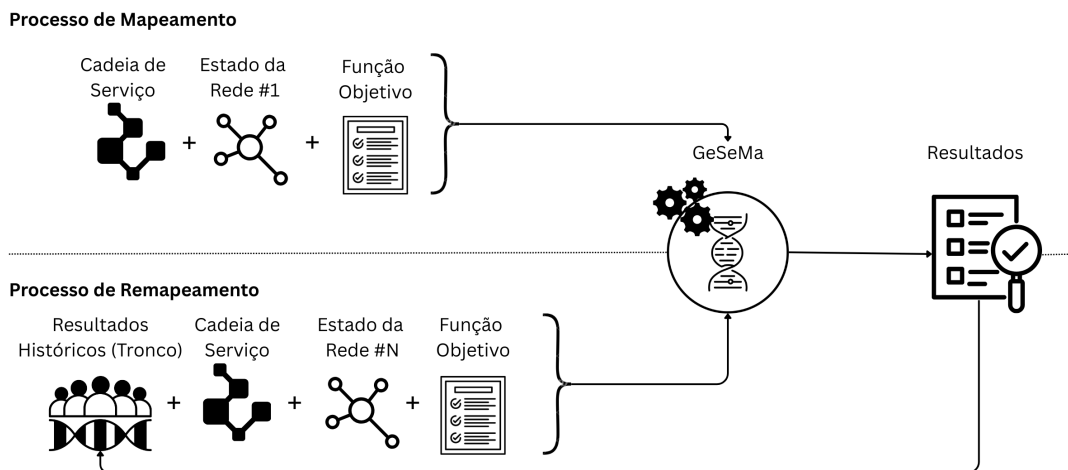
Após a execução do operador de Tronco, podem ocorrer dois cenários distintos: (i) o número de indivíduos inseridos na primeira geração é igual ao tamanho definido para a população inicial, o que dispensa a execução do operador de produção; ou (ii) o número de indivíduos inseridos é inferior ao tamanho esperado da população inicial, caso em que o operador de produção é executado até que esse tamanho seja atingido. Independentemente do cenário, após serem atendidos os requisitos de inicialização da população, o processo de otimização baseado na evolução de gerações é iniciado, e o operador de Tronco, tradicionalmente, não é mais utilizado.

## 5. Implementação e Avaliação Experimental

Esta seção visa avaliar o impacto do operador Tronco no GeSeMa aplicado ao remapeamento de serviços virtualizados de rede. A Seção 5.1 descreve a implementação e a integração do Tronco ao GeSeMa; a Seção 5.2 detalha o ambiente e as configurações experimentais; e a Seção 5.3 apresenta e discute os resultados obtidos.

### 5.1. Integração ao GeSeMa

O operador de Tronco foi implementado e integrado à solução GeSeMa [Fulber-Garcia et al. 2023] para a verificação de sua viabilidade técnica. Destaca-se a opção pelo GeSeMa dentre as alternativas disponíveis por dois motivos principais: (i) a possibilidade de definição da função objetivo por parte do usuário, expandindo as oportunidades de uso da solução e, por consequência, da implementação do operador de Tronco realizada; e (ii) a disponibilidade de um mecanismo de persistência de saídas com os resultados pertencentes à fronteira de Pareto relativa retornada pela solução na resolução de um problema, fornecendo as informações necessárias para a utilização do operador de Tronco de maneira nativa. Por fim, reafirma-se que o operador de Tronco é um operador genético genérico e pode ser integrado a qualquer algoritmo genético, desde que adaptado para atender às suas particularidades.



**Figura 1. Fluxo de Execução do GeSeMa: Mapeamento e Remapeamento**

Conforme exposto, o operador de Tronco foi implementado utilizando diretamente o arquivo de saída do GeSeMa como entrada. Esse arquivo, gerado com a opção “-o” na linha de comando, contém os melhores resultados para o mapeamento de um serviço, incluindo o resultado linearizado (cadeias ramificadas são suportadas, mas linearizadas para processamento, mantendo apenas os índices de conexão entre funções) e as avaliações correspondentes às métricas da função objetivo. O operador de Tronco, acionado pela opção “-i” na linha de comando, recebe esse arquivo, descarta as avaliações anteriores, valida os resultados históricos para o estado da rede no momento do remapeamento e os adapta às estruturas de dados do algoritmo genético, inserindo-os na população inicial. A Figura 1 ilustra os processos de mapeamento e remapeamento do GeSeMa.

Destaca-se, também, a possibilidade de concatenar arquivos de resultados de múltiplas execuções de mapeamento e remapeamento de um mesmo serviço, formando

uma janela histórica em um único arquivo de entrada. O operador de Tronco implementado no GeSeMa insere os resultados históricos (indivíduos) na população inicial conforme a ordem do arquivo de entrada, até que o tamanho da população seja atingido ou os resultados se esgotem, permitindo ao usuário definir a prioridade de inserção, *e.g.*, com base em quão recentes são ou em melhores avaliações históricas para uma métrica específica.

## 5.2. Ambiente e Contexto

Os testes foram executados em um ambiente computacional composto por um processador Intel Core i7-1355U, 16 GB de RAM (DDR4, 3.200 MT/s), executando Ubuntu 24.04.3 LTS (kernel 6.14.0-37-generic) e Python 3.12.3. O contexto de rede utilizado como referência nos experimentos foi construído com 35 domínios, representando os locais potenciais para o mapeamento das funções de um serviço virtualizado de rede, o qual é composto por uma cadeia linear de 11 funções virtualizadas de rede, implementando um serviço hierárquico de *cache* multimídia [Fulber-Garcia et al. 2023].

Cada domínio possui restrições de recursos computacionais, incluindo o número de interfaces de rede, memória e vCPUs disponíveis. De forma análoga, cada função virtualizada apresenta requisitos mínimos referentes aos mesmos recursos, os quais devem ser respeitados durante os processos de mapeamento e remapeamento. Duas métricas principais foram consideradas no processo de otimização. A primeira diz respeito ao custo local de implantação de uma função em um domínio, representando o custo computacional ou financeiro associado à alocação; e a segunda refere-se à largura de banda disponível entre os domínios, refletindo a capacidade da rede de suportar a comunicação entre funções de rede distribuídas.

A função objetivo foi configurada para minimizar o custo local de implantação e maximizar a largura de banda disponível entre os domínios selecionados. A avaliação das soluções foi realizada por meio da dominância de Pareto. Além disso, a meta-heurística genética utilizada foi a SPEA2, com a população inicial fixada em 100 indivíduos para todos os cenários avaliados. Foram comparadas duas abordagens principais: (i) Remapeamento “a Frio” (**RaF**), que corresponde à execução do GeSeMa sem o uso de resultados históricos; e (ii) Remapeamento com Tronco (**RcT**), no qual resultados históricos são incorporados à população inicial por meio do operador Tronco.

A seleção dos resultados históricos considerou uma janela de 30 execuções anteriores do GeSeMa, parametrizado para 30000 gerações cada. Dessas execuções, foram selecionados e agrupados exclusivamente os resultados associados à fronteira de Pareto de cada uma, ou seja, resultados não dominados. Foram avaliadas duas estratégias de seleção dos resultados históricos: (i) **randômica**, na qual as inserções no Tronco são escolhidas uniformemente entre os resultados elegíveis; e (ii) baseada nas **melhores fronteiras**, considerando os resultados nas fronteiras mais externas quando calculada a dominância de Pareto relativa entre eles. Além disso, o impacto da composição do Tronco foi avaliado por meio de duas proporções distintas na população inicial: **herança de 50%**, na qual metade da população inicial é composta por resultados históricos e a outra metade é gerada aleatoriamente; e **herança de 100%**, na qual toda a população inicial é formada por resultados históricos. Por fim, 3 cenários de teste são enumerados e descritos a seguir:

### 1. Mudanças graduais no estado da rede, remapeamento com o mesmo número

**de gerações:** execução do GeSeMa por 30000 gerações, considerando a ocorrência de mudanças no estado da rede que atingem 5% e 10% dos domínios, aprimorando em 5% e 10%, respectivamente, as métricas avaliadas, ou seja, a redução do custo de implantação e o aumento da largura de banda disponível. Esse cenário busca avaliar a influência do uso de resultados históricos em estados da rede gradualmente mais favoráveis;

2. **Mudanças disruptivas no estado da rede, remapeamento executado até a convergência:** execução do GeSeMa até a convergência (nenhuma mudança na fronteira de Pareto após 1500 gerações, quantidade denominada **passo geracional**), considerando modificações no estado da rede que atingem 50% e 75% dos domínios. Para cada caso, foram aplicados aprimoramentos aleatórios, entre 10% e 30%, nas métricas avaliadas relacionadas a cada domínio, permitindo avaliar a robustez do uso do Tronco frente a grandes mudanças no estado da rede;
3. **Mudanças graduais no estado da rede, mas que atingem diretamente os domínios presentes nos resultados históricos:** execução do GeSeMa por 30000 gerações, considerando a ocorrência de mudanças no estado da rede que causam 10% de degradação nas métricas avaliadas, caracterizada pelo aumento do custo de implantação e pela redução da largura de banda interdomínio disponível. A degradação é aplicada em 4 e 6 domínios presentes nos resultados históricos inseridos no Tronco, visando avaliar o impacto do operador em condições adversas.

### 5.3. Resultados e Discussões

Os resultados a seguir apresentam a média das fronteiras relacionadas à dominância de Pareto relativa calculada para os resultados provenientes dos processos de RaF e de RcT (ou seja, **quanto mais próximo de zero, melhor**), considerando um intervalo de confiança de 95% para 30 execuções.

As Figuras 2a e 2b refletem a execução dos testes do cenário 1, com, respectivamente, 5% e 10% dos domínios afetados no novo estado da rede. De modo geral, observa-se uma redução média de 45,65% nas fronteiras relativas às quais pertencem os resultados obtidos quando o operador Tronco é utilizado para o remapeamento, em comparação com quando não é utilizado. Essa redução demonstra que o GeSeMa é capaz de gerar melhores resultados quando, diante de uma mudança no estado da rede, o Tronco é aplicado. Ou seja, os resultados gerados a partir do Tronco apresentam menores custos de implantação e maiores larguras de banda disponíveis interdomínio.

Representados nas Figuras 3 e 4, os testes relacionados ao cenário 2 evidenciam reduções de 82,15% e 80,62% na média das fronteiras relativas dos resultados de remapeamento com o operador Tronco, considerando mudanças no estado da rede observadas em 50% e 75% dos domínios disponíveis, respectivamente. Do ponto de vista da quantidade de gerações necessárias para alcançar o critério de convergência, o RaF executa, aproximadamente, 3,7 e 2,9 vezes mais passos geracionais nos mesmos casos, conforme indicado nas Figuras 4a e 4b.

Os resultados associados aos testes do cenário 3, apresentados na Figura 5, seguem a mesma tendência daqueles obtidos nos cenários anteriores. Quando os domínios, especificamente aqueles presentes no conjunto de resultados históricos inseridos no Tronco, sofrem uma degradação de 10% em suas métricas (*i.e.*, aumento dos custos de recursos

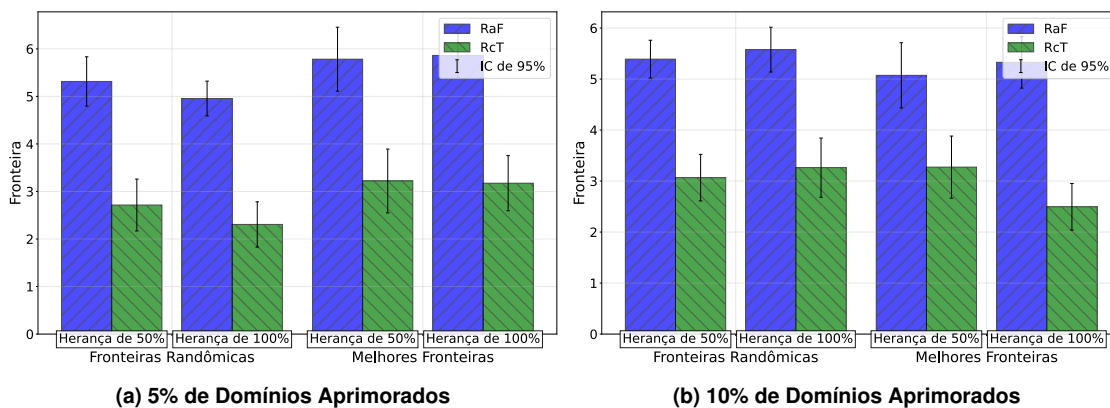


Figura 2. Cenário 1: Média das Fronteiras Relativas

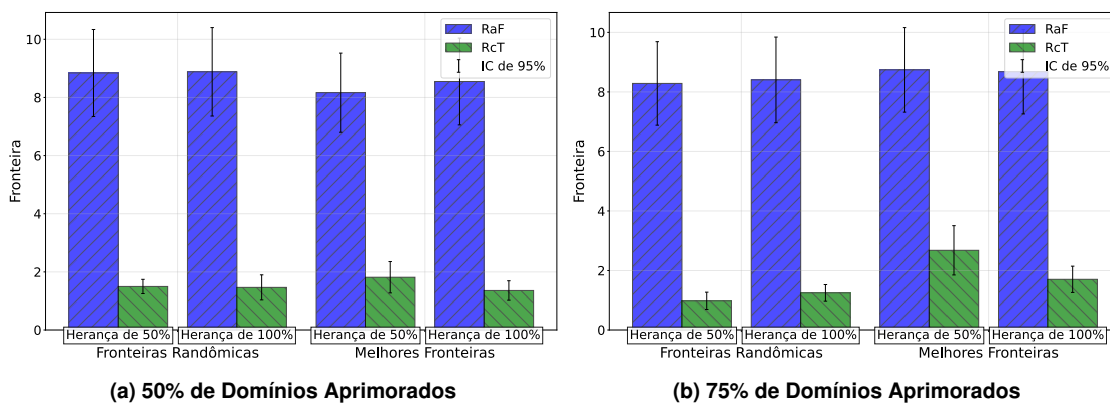


Figura 3. Cenário 2: Média das Fronteiras Relativas

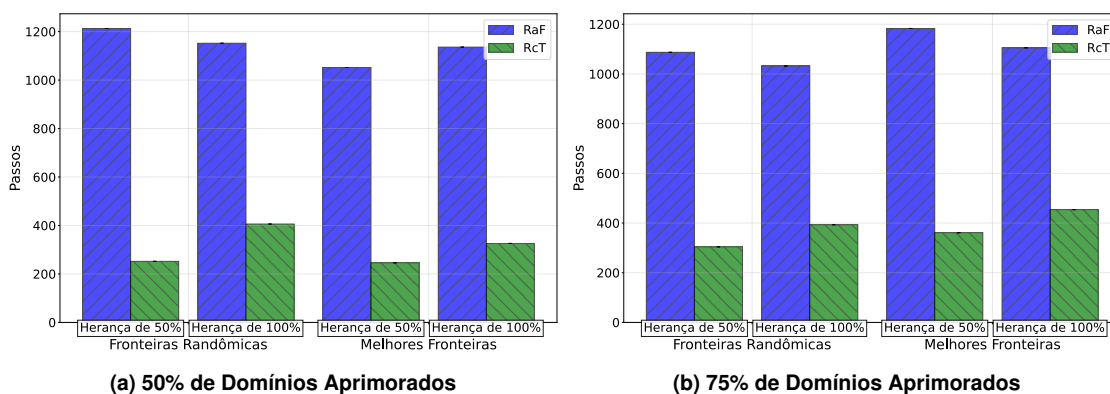


Figura 4. Cenário 2: Média de Passos Geracionais para a Convergência

computacionais e redução da largura de banda interdomínio disponível), as melhorias nos resultados de remapeamento com o Tronco se mantêm: reduções de 44,69% e 46,64% na média das fronteiras relativas (sendo 4 e 6 domínios degradados, respectivamente), quando comparadas aos resultados do RaF.

Quanto ao consumo de recursos computacionais (CPU e RAM), não se observam diferenças significativas nos picos de uso durante o remapeamento, independentemente da utilização do operador Tronco, conforme ilustrado na Figura 6, obtida para o cenário

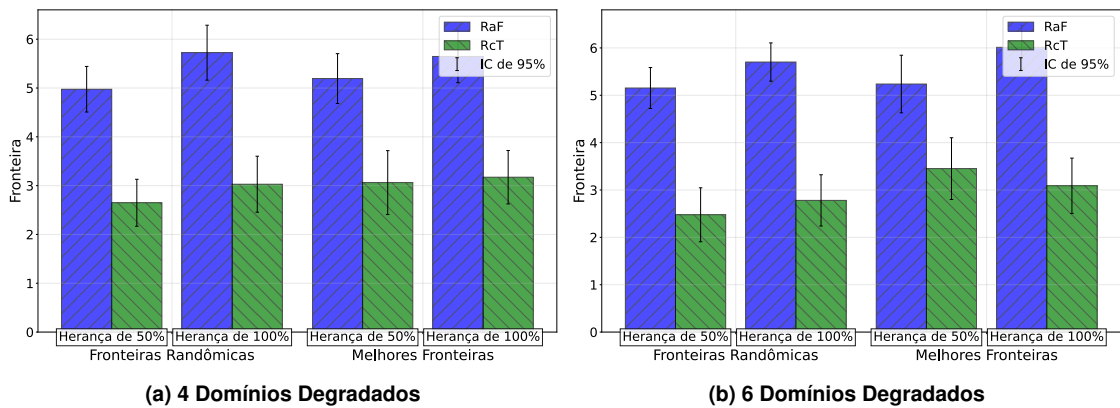


Figura 5. Cenário 3: Média das Fronteiras Relativas

1 (10% de aprimoramento em 10% dos domínios) executado até a convergência. Isso ocorre porque o Tronco atua efetivamente apenas no início do processo de otimização, na definição da população inicial, e, a partir desse ponto, todos os processos executados são idênticos tanto para o RaF quanto para o RcT. É importante notar, no entanto, que há uma diferença significativa no tempo médio de execução: 8,03 segundos para o RaF e 3,12 segundos para o RcT. Como o custo dos passos geracionais é equivalente em ambas as abordagens, essa diferença no tempo de execução reflete diretamente a quantidade média de passos geracionais necessários para alcançar a convergência.

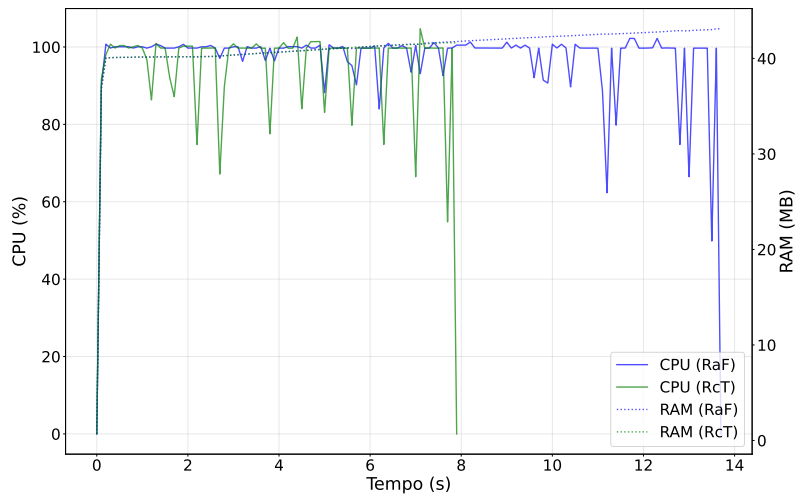


Figura 6. Cenário 1 (10%): Consumo de CPU e RAM até a Convergência

Os resultados apresentados ao longo desta seção demonstram que o operador Tronco impacta positivamente os processos de remapeamento, independentemente do cenário de teste considerado. Isso ocorre porque, mesmo em eventos de mudança no estado da rede mais acentuados, como os testados no segundo cenário, segmentos de resultados de mapeamentos ótimos (locais ou globais) em estados anteriores da rede continuam sendo promissores para o estado atualizado. Esses resultados históricos podem, então, (i) ser incorporados diretamente como resultados a serem mantidos no estado atualizado da rede (por elitismo, por exemplo) ou (ii) prover uma base promissora para o início do processo de otimização, com indivíduos contendo segmentos altamente adequados para

compor parcialmente resultados ótimos após a aplicação dos operadores genéticos.

## 6. Conclusão

É fato que o paradigma NFV flexibiliza o gerenciamento de funções e serviços de rede, especialmente no que se refere à otimização de sua implantação. Como consequência, surge a necessidade de elaborar soluções de tomada de decisão pertinentes ao mapeamento de serviços em múltiplos domínios e pontos de presença, sendo os algoritmos genéticos multiobjetivo alternativas adequadas a esse processo. Apesar de reconhecerem a vivacidade da rede, soluções genéticas de mapeamento não utilizam resultados históricos para orientar o remapeamento de um serviço quando ocorre uma mudança no estado da rede.

Nesse contexto, este artigo apresentou o operador genético de Tronco, que manipula a população inicial, compondo-a com resultados históricos, visando otimizar o remapeamento de serviços de rede. Os experimentos demonstram que o operador proposto é efetivo em diversos cenários de teste quando integrado à solução GeSeMa. Isso se deve à condução prévia do processo de otimização a regiões promissoras do espaço de busca, tornando o algoritmo mais assertivo na busca por domínios ótimos para o mapeamento de funções de uma cadeia de serviço.

Trabalhos futuros incluem: determinar a influência da temporalidade no uso de resultados históricos; analisar a resistência do operador de tronco a interrupções no estado da rede; avaliar a possibilidade de utilizar resultados históricos de outros serviços com características semelhantes no operador Tronco; e investigar métodos mais eficientes de definição do estado da rede que permitam disparar automaticamente processos de remapeamento de serviços virtualizados de rede.

## Disponibilidade de Artefatos

Solução GeSeMa com o operador de Tronco: <https://tinyurl.com/GeSeMa>.  
Experimentos e resultados: <https://tinyurl.com/StemOperator>.

## Agradecimentos

Este trabalho foi desenvolvido com o suporte da Universidade Federal do Paraná (COFPI/PRPI 19/2025 - 23075.058047/2025-51), PROEX CAPES (AUXPE 88881.189840/2025-01) e CNPq (305108/2025/5).

## Referências

- Benzekki, K. et al. (2016). Software-defined networking (sdn): a survey. *Security and communication networks*, 9(18):5803–5833.
- De Falco, I. et al. (2002). Mutation-based genetic algorithm: performance evaluation. *Applied Soft Computing*, 1(4):285–299.
- Forrest, S. (1996). Genetic algorithms. *ACM Computing Surveys*, 28(1):77–80.
- Fulber-Garcia, V. et al. (2020). Network service topology: Formalization, taxonomy and the custom specification model. *Computer Networks*, 178:107337.
- Fulber-Garcia, V. et al. (2023). Customizable mapping of virtualized network services in multi-datacenter environments based on genetic metaheuristics. *Journal of Network and Systems Management*, 31(4):71.

- Fulber-Garcia, V. et al. (2024a). Breaking the limits: Bio-inspired sfc deployment across multiple domains, clouds and orchestrators. In *IEEE Conf. on Network Function Virtualization and Software Defined Networks*, pages 1–6.
- Fulber-Garcia, V. et al. (2024b). Uma estratégia bioinspirada para alocação dinâmica de sfcs em múltiplos domínios, nuvens e orquestradores nfv. In *Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos*, pages 183–196.
- Han, S. et al. (2020). Virtualization in programmable data plane: A survey and open challenges. *IEEE Open Journal of the Communications Society*, 1:527–534.
- Herrera, J. G. and Botero, J. F. (2016). Resource allocation in nfv: A comprehensive survey. *IEEE Trans. on Network and Service Management*, 13(3):518–532.
- Huff, A. et al. (2020). Building multi-domain service function chains based on multiple nfv orchestrators. In *IEEE Conf. on Network Function Virtualization and Software Defined Networks*, pages 19–24.
- Lambora, A. et al. (2019). Genetic algorithm-a literature review. In *Int. Conf. on Machine Learning, Big Data, Cloud and Parallel Computing*, pages 380–384.
- Li, Y. et al. (2020). Cost-and-qos-based nfv service function chain mapping mechanism. In *IEEE/IFIP Network Operations and Management Symposium*, pages 1–9.
- Liu, J. et al. (2024). Multi-objective model and genetic algorithm for multisource multi-cast vnf service chain deployment problem. *Int. Journal of Computer Science*, 51(6).
- Oliveira, G. W. et al. (2026). Observability: The missing piece of management in nfv-based network environments. In *International Workshop on ADVANCEs in ICT Infrastructures and Services*.
- Pachau, J. L. et al. (2021). An overview of crossover techniques in genetic algorithm. *Modeling, Simulation and Optimization*, pages 581–598.
- Rankothge, W. et al. (2015). Experimental results on the use of genetic algorithms for scaling virtualized network functions. In *Conf. on Network Function Virtualization and Software Defined Network*, pages 47–53.
- Rehman, A. et al. (2019). Network functions virtualization: The long road to commercial deployments. *IEEE Access*, 7:60439–60464.
- Rodis, P. and Papadimitriou, P. (2021). Intelligent network service embedding using genetic algorithms. In *IEEE Symp. on Computers and Communications*, pages 1–7.
- Shukla, A. et al. (2015). Comparative review of selection techniques in genetic algorithm. In *Int. Conf. on Futuristic Trends on Computational Analysis and Knowledge Management*, pages 515–519.
- Souza, R. et al. (2020). Nfv data centers: A systematic review. *IEEE Access*, 8:51713–51735.
- Voorneveld, M. (2003). Characterization of pareto dominance. *Operations Research Letters*, 31(1):7–11.
- Ward, P. S. (2010). Taxonomy, phylogenetics, and evolution. *Ant ecology*, 1.
- Zhang, T. et al. (2020). Nfv platforms: Taxonomy, design choices and future challenges. *IEEE Trans. on Network and Service Management*, 18(1):30–48.