

Mapeamento Customizado de Serviços de Rede em Múltiplos Domínios Baseado em Heurísticas Genéticas

Vinicius Fulber-Garcia¹, Carlos R. Paula dos Santos²,
Eduardo J. Spinosa¹, Elias P. Duarte Júnior¹

¹Universidade Federal do Paraná (UFPR)
Caixa Postal 19018 81531-990, Curitiba - PR

²Universidade Federal de Santa Maria (UFSM)
Av. Roraima, 1000 - Camobi, Santa Maria - RS

vfgarcia@inf.ufpr.br, csantos@inf.ufsm.br,
{spinosa, elias}@inf.ufpr.br

Abstract. *The deployment of network services is one of the major tasks of the Network Function Virtualization (NFV) paradigm. However, current multi-domain mapping solutions present several constraints regarding the adopted optimization models and evaluation metrics. This inflexibility ultimately leads to sub-optimized mappings that do not meet the requirements of the parties involved in the process (e.g., network operators, clients, providers, etc.). This work proposes a new mapping solution based on genetic heuristics. The new solution enables the configuration on-demand of the evaluation criteria used to generate candidate mappings. The solution takes into consideration different policies, technological constraints, and evaluation metrics which are specified individually, for each request document. Finally, we demonstrate the feasibility of the proposed solution (convergence and execution time) through a case study.*

Resumo. *A implantação de serviços de rede é um dos principais tarefas do paradigma de Virtualização de Funções de Rede (Network Function Virtualization - NFV). As soluções de mapeamento multi-domínio existentes apresentam diversas restrições em termos do modelos de otimização e métricas de avaliação que adotam. Essa inflexibilidade leva, em última instância, a mapeamentos subotimizados que não correspondem exatamente às necessidades das partes envolvidas no processo, incluindo operadores de rede, clientes, provedores, entre outros. Este trabalho propõe uma nova solução de mapeamento baseada em heurísticas genéticas. Essa nova solução permite a configuração sob demanda dos critérios de avaliação utilizados para a geração de mapeamentos candidatos, sendo capaz de considerar diferentes políticas, limitações tecnológicas e métricas de avaliação definidas de forma específica para cada requisição. Por fim, demonstramos a viabilidade da solução proposta (convergência e tempo de execução) através de um estudo de caso.*

1. Introdução

Infraestruturas de comunicação modernas dependem de diversas funções de rede para operar (e.g., roteadores, filtros, balanceadores de carga, sistemas de detecção de intrusão, entre tantos outros). Tradicionalmente, a maior parte dessas funções são implementadas

através de equipamentos dedicados e proprietários – chamados de *physical appliances*. Apesar de alcançarem alto desempenho no processamento de pacotes, esses equipamentos dedicados apresentam alto custo operacional e de capital, baixa escalabilidade e mobilidade, além de processos complexos de manutenção [NFVISG 2012]. Essas características aprofundam o problema conhecido como ossificação da Internet [Handley 2006]. Nesse contexto, o paradigma de Virtualização de Funções de Rede (*Network Functions Virtualization* - NFV) surge como uma alternativa para contornar as dificuldades citadas. O paradigma de NFV utiliza técnicas de virtualização para fornecer funções de rede através de dispositivos virtuais que executam em hardware genérico (*e.g.*, arquitetura x86). Diversos esforços da indústria e da academia apresentam inúmeras vantagens de NFV em comparação com redes tradicionais, tais como a simplificação dos processos de desenvolvimento, implantação e manutenção das funções de rede, a redução de custos e a flexibilidade para provisionamento de funções inovadoras de rede.

Em uma rede em produção são normalmente utilizados diversos serviços para processar o tráfego de dados dos usuários, com as mais diversas funcionalidades. No paradigma de NFV, a um conjunto de funções orquestradas para oferecer um serviço de rede, dá-se o nome de *Service Function Chain* (SFC, que pode ser traduzido como Cadeia de Função de Serviço) [Quinn et al. 2015]. Uma das principais tarefas relacionadas à operação de serviços de rede virtualizados é a sua implantação no substrato físico. Esse processo, denominado Alocação de Recursos em NFV (*NFV Resource Allocation* – NFV-RA) [Herrera et al. 2016], é realizado através de três etapas principais: composição (*composition*), integração (*embedding*) e programação da execução (*scheduling*). Dada a sua importância, os desafios decorrentes de cada uma dessas tarefas vêm sendo extensivamente tratados na literatura recente [Gil-Herrera et al. 2017, Fulber-Garcia et al. 2020, Wang et al. 2017, Even et al. 2019].

Soluções tradicionais de implantação de NFV são baseadas em técnicas de otimização, usualmente baseadas em um conjunto de critérios (*i.e.*, métricas) predefinidos e imutáveis, não permitindo customizações [Herrera et al. 2016]. Essa característica faz com que os interessados em implantar um serviço de rede (*e.g.*, clientes, provedores, operadores de rede) tenham que adaptar suas necessidades às soluções existentes, o que pode gerar resultados inadequados. Não diferente, essas limitações de customização estão presentes nas soluções que realizam o mapeamento de serviços em múltiplos domínios [Dietrich et al. 2015, Abujoda et al. 2016, Zhang et al. 2016, Riera et al. 2016, Wang et al. 2017], que em geral permitem apenas ajustes simplificados (*e.g.*, ponderação de cada métrica). Entre as métricas e objetivos de otimização dessas soluções pode-se citar: minimização de custos financeiros, minimização de sobrecarga na rede, minimização do atraso interdomínio, maximização ou minimização da quantidade de domínios utilizados e estabilização do balanceamento de carga. Apesar de formarem um conjunto significativo, essas métricas podem não representar por completo as necessidades pontuais das partes interessadas na implantação.

Nesse contexto, é importante observar que, além dos objetivos e métricas de otimização, existem outros fatores que podem influenciar o mapeamento de serviços de rede em múltiplos domínios, entre eles: a diversidade das infraestruturas (*e.g.*, particular, pública, conexões criptografadas ou não, protocolos de comunicação diversos); diferentes tipos de serviços (*e.g.*, roteamento, entrega de conteúdo, segurança); e suas topologias

(*i.e.*, linear e ramificada). Esses fatores também são comumente (de forma completa ou parcial) negligenciados pelas soluções existentes, agravando a inflexibilidade das mesmas e limitando ainda mais as suas aplicações.

Considerando o cenário apresentado, este trabalho propõe o *Genetic Service Mapping* (GeSeMa), uma nova solução para o mapeamento de serviços de rede virtualizados em ambientes multidomínio. Diferente das soluções atuais de implantação, a solução GeSeMa realiza uma avaliação customizada a partir de métricas e objetivos de otimização aplicados sobre uma topologia de serviço e uma infraestrutura de rede, todos informados pelo usuário através de um modelo de requisição próprio. Além disso, a solução proposta emprega métodos de otimização multiobjetivo baseados em heurísticas genéticas para determinar resultados adequados às requisições providas em tempo factível. O desempenho da solução é avaliado através de um estudo de caso.

O restante deste artigo está organizado como se segue: a Seção 2 apresenta a fundamentação teórica e trabalhos relacionados; a Seção 3 descreve o funcionamento da nova solução proposta em detalhes; a Seção 4 apresenta uma avaliação através de estudo de caso; finalmente, a Seção 5 conclui o artigo e expõe perspectivas futuras em relação ao trabalho.

2. Fundamentação Teórica e Trabalhos Relacionados

Nesta seção são apresentados conceitos que fundamentam o presente trabalho, além dos principais trabalhos que abordam a técnica de mapeamento multidomínio na integração de serviços de rede virtualizados.

2.1. Implantação de Serviços de Rede Virtualizados

O paradigma NFV prevê a criação de serviços complexos através do encadeamento de múltiplas funções [Quinn et al. 2015]. Serviços de rede são construídos a partir de uma estrutura chamada SFC que contém tanto informações de identificação das funções requisitadas quanto a topologia do serviço (*i.e.*, grafo de encaminhamento) representando suas interconexões. Esses serviços são implantados em um ambiente virtualizado através do processo conhecido como NFV-RA [Herrera et al. 2016]. O NFV-RA, por sua vez, consiste de três tarefas de otimização: (i) **composição** da topologia de serviço; (ii) **integração** do serviço no ambiente virtualizado; (iii) **programação da execução** das funções de rede nos servidores utilizados. Recentemente, os desafios particulares de cada uma dessas tarefas vêm sendo extensivamente explorados [Gil-Herrera et al. 2017, Fulber-Garcia et al. 2020, Wang et al. 2017, Even et al. 2019]. Com isso, objetiva-se determinar implantações que otimizem métricas prioritárias para o aperfeiçoamento da qualidade do serviço e cumprimento de políticas de execução.

Em especial, a tarefa de integração pode ser realizada por diferentes técnicas, selecionadas de acordo com o ambiente virtualizado disponível. As três técnicas mais comumente aplicadas são: mapeamento multidomínio (*mapping*); seleção de funções de rede (*selection*); e alocação em servidores (*placement*). O mapeamento multidomínio destina-se a determinar domínios administrativos e provedores capazes de suportar um serviço, ou parte dele, em sua infraestrutura. A seleção de funções de rede busca por funções já implantadas em um ambiente virtualizado, para utilizá-las de forma compartilhada entre diversos serviços. Finalmente, a alocação de servidores verifica, dentro de

um domínio já definido, quais máquinas físicas devem hospedar e executar determinadas funções de rede. Essas técnicas são continuamente adaptadas para suprir necessidade de provedores, operadores de rede e clientes. Atualmente, além de cumprir o papel de operacionalização do serviço, essas técnicas de integração também são exploradas como um modelo de negócio em arquiteturas de *Network-as-a-Service* [Boubendir et al. 2016]. Especificamente, o presente trabalho investiga novas possibilidades de integração através da técnica de mapeamento multidomínio.

2.2. Heurísticas Genéticas

Algoritmos genéticos se baseiam nos princípios Darwinianos de evolução das espécies para resolver uma série de problemas (*e.g.*, otimização, agrupamento, simulação, etc.) [Hasançebi et al. 2000]. Esses algoritmos analisam indivíduos representados através de um cromossoma (*i.e.*, um vetor, que representa uma possível solução) composto por um conjunto de genes (*i.e.*, características pontuais do problema), cada um determinado por um alelo (*i.e.*, um valor que um gene pode assumir). Os indivíduos são submetidos a um processo evolutivo, tipicamente baseado nas operações de cruzamento e mutação, de forma a obter novos indivíduos mais aptos a solucionar o problema. Nas operações de cruzamento, dois ou mais indivíduos são selecionados e seus cromossomas parcialmente combinados de forma a criar um novo indivíduo descendente. Já em operações de mutação, um indivíduo é submetido a uma mudança particular no alelo de um gene específico. É importante ressaltar que algoritmos genéticos são heurísticas e não garantem a obtenção do resultado ótimo global. Além disso, por executarem operações estocásticas, múltiplas execuções de uma mesma heurística genética podem retornar resultados heterogêneos entre si. Em especial, as heurísticas genéticas são adequadas para resolverem problemas com grande espaço de busca, obtendo soluções factíveis em um tempo de execução viável.

Quando utilizados para solucionar problemas de otimização, as heurísticas genéticas se subdividem em duas classes principais: mono-objetivo e multiobjetivo. Heurísticas dedicadas a problemas mono-objetivo visam otimizar apenas uma função ou métrica de avaliação, enquanto que aquelas desenvolvidas para problemas multiobjetivo determinam uma relação de custo-benefício entre múltiplas funções e métricas, normalmente definida através da análise das fronteiras de Pareto. Indivíduos em uma mesma fronteira não apresentam relação de dominância entre si, isto é, para cada indivíduo existe ao menos uma métrica ou função entre o conjunto analisado cujo resultado é melhor em relação aos demais. Também, problemas de otimização podem apresentar restrições quanto a regiões indisponíveis do espaço de busca e combinações de genes que geram indivíduos inválidos. Diversas técnicas podem ser utilizadas para tratar essas restrições: controle da população inicial, especificação de mutações inválidas, descarte de indivíduos, entre outros. No contexto deste trabalho são considerados problemas de otimização multiobjetivo com restrições e utilizados os algoritmos *Nondominated Sorting Genetic Algorithm II* (NSGAI) [Deb et al. 2002] e *Strength Pareto Evolutionary Algorithm 2* (SPEA2) [Zitzler et al. 2001].

2.3. Trabalhos Relacionados

Diferentes mapeamentos em múltiplos domínios são capazes de prover um serviço de rede com sucesso. Entretanto, considerando um conjunto de políticas, restrições

e métricas de interesse, o desempenho verificado em mapeamentos distintos é tipicamente diferente [Herrera et al. 2016]. Assim, soluções que visam determinar mapeamentos válidos enquanto qualificam os mesmos se tornam necessárias. Em geral, as soluções de mapeamento multidomínio podem ser divididas em duas classes principais: (i) centralizadas e (ii) distribuídas. Soluções centralizadas são executadas em um único ponto de processamento. Essas recuperam todas as informações necessárias relativas ao serviço requisitado e aos domínios disponíveis, as avaliam considerando um conjunto de métricas de interesse e retornam um ou mais mapeamentos adequados a elas. Por outro lado, soluções distribuídas espalham a requisição de integração dos elementos virtualizados entre os domínios disponíveis através de um protocolo de comunicação comum aos mesmos. Neste caso, cada domínio avalia a requisição para determinar segmentos que podem ser hospedados por eles, encaminhando o restante aos domínios vizinhos. Soluções centralizadas são propostas em [Dietrich et al. 2015, Riera et al. 2016, Wang et al. 2017], enquanto que soluções distribuídas são apresentadas em [Abujoda et al. 2016, Zhang et al. 2016].

Em [Dietrich et al. 2015] informações ofuscadas (*i.e.*, dados agregados através de uma peso) sobre a adequabilidade de alocação de funções de rede nos provedores de infraestrutura são utilizadas. Assim, essa solução procede a otimização do mapeamento do serviço considerando quatro critérios fixos: minimização dos (i) custos financeiros, (ii) do número de provedores e domínios diferentes utilizados, (iii) utilização de recursos computacionais e maximização da (iv) adequabilidade das funções de rede. Já em [Riera et al. 2016], uma plataforma de orquestração de serviços de rede multidomínio, chamada TeNOR, é proposta. A plataforma TeNOR conta com uma solução de mapeamento multidomínio nativa. Essa solução centraliza informações de custos financeiros, atraso de transmissão e utilização de recursos computacionais, que utiliza como entradas para sua política de minimização que decide o mapeamento das funções de rede de uma topologia serviço requisitada. Finalmente, em [Wang et al. 2017], o mapeamento ocorre em um cenário híbrido onde tanto domínios privados quanto de provedores disponibilizam recursos relacionados a redes ópticas. A solução proposta visa determinar mapeamentos que minimizem custos financeiros (privilegiando assim domínios privados) e a utilização de *slots* de frequência nos canais ópticos.

Na solução proposta em [Zhang et al. 2016] uma metodologia centrada em vértices é utilizada para realizar o mapeamento multidomínio. Essa solução realiza rodadas de trocas de mensagens entre provedores para determinar mapeamentos possíveis iterativamente. Além disso, uma política que limita a quantidade de alocações de função de rede a cada rodada de troca de mensagens é aplicada, evitando assim monopolização da topologia de serviço por um único provedor. Nessa solução, nenhuma otimização específica é realizada, sendo retornado ao usuário apenas um conjunto de mapeamentos possíveis que atendem aos requisitos de instanciação das funções de rede que compõem o serviço requisitado. Utilizando um processo semelhante ao apresentado em [Zhang et al. 2016], a solução DistNSE [Abujoda et al. 2016] determina mapeamentos candidatos de forma distribuída e os avalia considerando uma função de otimização que visa a minimização do custo financeiro total e balanceamento de carga interdomínio. Porém, diferente da solução em [Zhang et al. 2016], não existe limitação no número de alocações de funções de rede feitas por provedores a cada rodada de troca de mensagens.

A Tabela 1 sumariza as principais características das soluções apresentadas. Apesar destas soluções trabalharem com múltiplos objetivos de otimização, elas não permitem aos seus usuários configurar o conjunto de métricas que devem ser avaliadas. Essa ausência de suporte a customização acarreta dificuldades para modelar e avaliar políticas e acordos que são diretamente ligados à tarefa de implantação (*e.g.*, atraso máximo, distância geográfica máxima). Além disso, limitações quanto ao processamento de restrições são notadas nas soluções [Riera et al. 2016] e [Wang et al. 2017]. Nelas não é possível definir dependências de domínio específicas de algumas funções de rede, o que torna inviável, por exemplo, a utilização das mesmas para distribuir serviços híbridos onde há funções implementadas fisicamente em um nodo da rede coexistem com funções virtualizadas. Finalmente, nenhuma das heurísticas utilizadas pelas soluções existentes é categorizada como uma heurística genética. Nós argumentamos que a utilização de algoritmos genéticos, além de permitir a definição de resultados satisfatórios em tempo hábil para problemas complexos de mapeamento, pode facilitar a realocação de serviços quando mudanças no estado da rede ou nas características de alguns domínios ocorrem. Dessa forma, assumindo que tipicamente nem todos os domínios de rede mudam em um dado intervalo de tempo, uma nova execução da heurística pode se beneficiar de resultados anteriores como ponto inicial de sua busca, assim sugerindo a análise de mapeamentos iniciais que, com pequenas modificações, podem voltar a serem candidatos promissores à solução do problema.

Classe	Trabalho Referenciado				
	[Dietrich]	[Riera]	[Wang]	[Abujoda]	[Zhang]
Heurística	Centralizado	Centralizado	Centralizado	Distribuído	Distribuído
Métricas de Avaliação	Mapeamento de Subgrafo Custos Financeiros Quantidade de Provedores Recursos Computacionais Adequabilidade (Pesos)	N/A Custos Financeiros Atraso interdomínio Recursos Computacionais	<i>K-cut</i> Slots de Frequência Custos Operacionais	<i>Time to Live</i> Custos Financeiros Balanceamento de Carga	N/A
Suporte a Customização de Métricas de Avaliação	×	×	×	×	×
Suporte a Definição de Dependências de Domínio [Fulber-Garcia et al. 2020]	✓	×	×	✓	✓

Tabela 1. Sumarização de Características das Soluções de Mapeamento

Também são destacadas soluções de implantação que, em outros contextos, utilizam heurísticas genéticas para resolver problemas de otimização. Em [Cao et al. 2016], [Carpio et al. 2017], [Khebbache et al. 2018] e [Tavakoli-Someh et al. 2019] são apresentadas soluções baseadas em algoritmos genéticos para executar a tarefa de alocação de servidores durante a integração de um serviço de rede virtualizado. Em especial, a solução em [Cao et al. 2016] utiliza soluções baseadas em *Multi-Objective Genetic Algorithm* (MOGA) e NSGAI para otimizar a seleção de servidores considerando dois objetivos: redução da sobrecarga das conexões entre servidores e balanceamento de carga entre servidores. Já na solução proposta em [Carpio et al. 2017], uma heurística genética é construída visando a otimização da seleção de servidores em múltiplos centros de processamento. Essa solução tem por objetivo minimizar a sobrecarga das conexões entre os servidores escolhidos. Similarmente, em [Khebbache et al. 2018], o NSGAI é utilizado para minimizar dois objetivos: o número de servidores utilizados e a sobrecarga das conexões entre eles. Finalmente, em [Tavakoli-Someh et al. 2019], uma solução é proposta também sobre NSGAI visando a maximização do uso médio dos servidores disponíveis e, ao mesmo tempo, a minimização do número de servidores necessários para a integração do serviço. Em um segundo contexto, de programação da execução de funções de rede

virtualizadas, uma solução baseada em NSGAII é proposta visando otimizar o uso de recursos computacionais de servidores hospedeiros e o tempo de conclusão do processamento das funções de rede neles hospedadas [Ma et al. 2017]. Apesar de utilizarem heurísticas genéticas, nenhuma das soluções apresentadas é destinada ao mapeamento multidomínio de serviços de rede, além de não serem capazes de otimizar métricas de interesse customizadas pelo usuário.

3. *Genetic Service Mapping*

A solução *Genetic Service Mapping*¹ (GeSeMa) baseia-se em heurísticas genéticas para realizar o mapeamento de serviços de rede em ambientes com múltiplos domínios. O objetivo é oferecer uma solução flexível e customizável, que permita ao usuário definir não apenas o serviço e suas dependências mas também as métricas e objetivos de otimização que devem ser avaliados. As informações são providas ao GeSeMa através de um modelo de requisição estruturado pela linguagem de serialização de dados *YAML Ain't Markup Language* (YAML). Esta seção apresenta o modelo de requisição e a metodologia de mapeamento multidomínio adotada pela solução proposta.

3.1. Modelo de Requisição

O modelo de requisição da solução GeSeMa, ilustrado na Figura 1, apresenta três objetos principais que especificam: (i) a estrutura do serviço a ser mapeado e de suas funções de rede (*SERVIÇO*); (ii) as métricas, restrições e objetivos de otimização do mapeamento (*MÉTRICAS*); e (iii) os domínios disponíveis e suas características individuais (*DOMÍNIOS*). Particularmente, o objeto de *SERVIÇO* carrega a especificação da topologia do serviço de rede requisitado. Essa topologia, por sua vez, é representada através de uma cadeia de caracteres especificada seguindo as regras sintáticas e semânticas do modelo *Service ChAin Grammar* [Garcia et al. 2019]. Da mesma forma, o objeto *SERVIÇO* especifica os requisitos computacionais de cada uma das funções de rede utilizadas na topologia do serviço, estes dados estão incluídos no subobjeto *FUNÇÕES*. É importante observar que para cada função especificada na topologia do serviço existe também uma entrada correspondente no subobjeto *FUNÇÕES*. Essa entrada, registrada com o identificador da função, contém dados relativos aos requisitos mínimos da função como, por exemplo, memória, núcleos de processamento virtuais e interfaces de rede virtuais, todos estes definidos por valores positivos do tipo inteiro.

O objeto *MÉTRICAS* especifica critérios utilizados pela heurística genética do GeSeMa na busca, avaliação e otimização de candidatos a resolver a requisição de mapeamento. As métricas empregadas na solução proposta podem ser de duas categorias: local e transição. Métricas locais são avaliadas no momento da alocação de uma função de rede em um dado domínio (*e.g.*, custo da alocação, sobrecarga do domínio, etc.), enquanto métricas de transição são utilizadas somente na ocorrência de uma mudança de domínio (*e.g.*, atraso, saltos, distância geográfica, etc.). Essas categorias são representadas no documento de requisição através dos subobjetos *LOCAL* e *TRANSIÇÃO*. Dessa forma, cada um desses subobjetos podem conter múltiplas métricas. Cada métrica deve ser unicamente identificada (*ID*), além de apresentar dois atributos obrigatórios: *OBJETIVO* e *RESTRIÇÕES*. Os objetivos declaram como a métrica deve ser computada em busca

¹Disponível em <https://github.com/ViniGarcia/NFV-FLERAS>

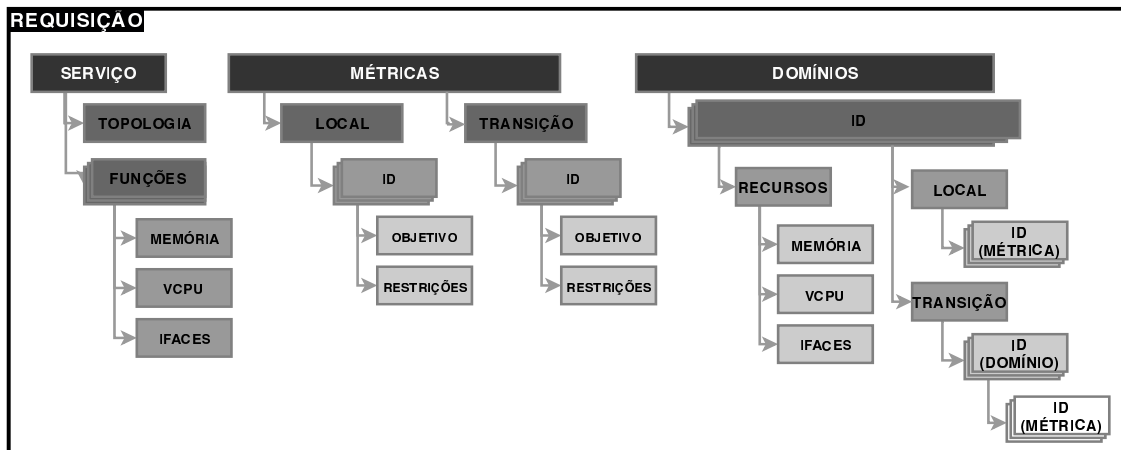


Figura 1. Modelo de Requisição GeSeMa

de resultados ótimos. Isso ocorre ao definir o problema como uma MAXIMIZAÇÃO ou MINIMIZAÇÃO. Finalmente, as restrições consistem de uma lista de cadeias de caracteres que definem limites de aceitação para o valor de uma métrica. Para isso os operadores relacionais "<", ">", "<=", ">=", "==" e "!=" podem ser utilizados em conjuntos a valores reais ou inteiros (e.g., "< 30", "== 0", ">= 15, 5").

Por fim, o objeto DOMÍNIOS especifica as infraestruturas físicas e virtuais disponíveis e suas transições, gerando um grafo direcionado, no qual os vértices carregam informações de métricas locais e as arestas armazenam dados relativos às métricas de transição. Um domínio particular é caracterizado por um subobjeto com identificador único (ID). Cada subobjeto de domínio, por sua vez, contempla três outros subobjetos: RECURSOS, LOCAL e TRANSIÇÃO. O subobjeto de RECURSOS informa a disponibilidade do domínio em relação à memória (MEMÓRIA), núcleos virtuais de processamento (VCPU) e interfaces virtuais de rede (IFACES). Já os subobjetos LOCAL e TRANSIÇÃO são utilizados na declaração de valores de *benchmarking* (ou adquiridos em catálogos de provedores de infraestrutura) que são utilizados no processo de otimização. Esses subobjetos têm forte relação com o objeto de MÉTRICAS, devendo haver correspondência completa entre os identificadores das métricas declaradas e os identificadores dos *benchmarks* fornecidos. Em especial, cada entrada no subobjeto de TRANSIÇÃO declara, primeiramente, para qual domínio a transição ocorre (utilizando o identificador do mesmo) e só então define os valores das métricas a serem utilizadas na otimização da transição propriamente dita.

3.2. Metodologia Genética de Mapeamento Multidomínio

A solução GeSeMa é capaz de executar dois algoritmos genéticos amplamente utilizados: NSGAI [Deb et al. 2002] e SPEA2 [Zitzler et al. 2001]. A decisão por um dos algoritmos cabe ao usuário, este podendo usufruir de qualidades específicas dos mesmos que melhor se adequam às peculiaridades do serviço de rede e aos objetivos do mapeamento requisitado. As principais características da solução proposta são sumarizadas a seguir:

1. **Indivíduos:** Cada indivíduo apresenta um cromossoma modelado através de um vetor de tamanho N , onde $N > 1$ equivale à quantidade de funções de rede que compõem o serviço (i.e., cada função de rede é mapeada para uma posição do

vetor). Os genes consistem de números inteiros com valor (*i.e.*, alelo) presente no intervalo de indexadores $[0, M - 1]$, sendo $M > 0$ a quantidade de domínios disponíveis para a realização do mapeamento.

2. **População:** A população inicial é criada de maneira aleatória, sendo garantido apenas que nenhum dos indivíduos gerados viola dependências de domínios particulares (*i.e.*, fixando o valor do domínio presente na dependência na posição do vetor representante da função que a carrega). O tamanho da população é um parâmetro decidido pelo usuário.
3. **Objetivos e restrições:** Objetivos e restrições (políticas, topologia de rede, recursos computacionais e dependências de domínio) são avaliados para todos os indivíduos em cada geração. Na ocorrência de um indivíduo inválido, este é registrado em uma lista tabu, evitando reavaliações em uma eventual nova ocorrência do mesmo. Além disso, mecanismos de recuperação de indivíduos inválidos também são utilizados (discutidos posteriormente).
4. **Seleção:** A seleção de indivíduos para passarem pelo processo de cruzamento é realizada através do mecanismo de torneio. Nesse mecanismo I indivíduos são selecionados aleatoriamente e aquele presente na fronteira de Pareto mais externa é escolhido. O tamanho do torneio $I > 1$ é determinado pelo usuário.
5. **Cruzamento:** Quatro operadores de cruzamento são fornecidos aos usuários: *Simulated Binary Crossover*, *Half Uniform Crossover*, *Partially Mapped Crossover* e *Subtour Selection Crossover*. Cada um apresenta comportamentos operacionais diferentes, sendo que a escolha por um deles deve considerar as particularidades de cada requisição [Hasançebi et al. 2000]. A taxa de cruzamento (*i.e.*, probabilidade de aplicação do operador) por geração também pode ser configurada pelo usuário.
6. **Mutação:** Dois operadores de mutação são providos: mutação por redefinição e mutação por troca. A redefinição escolhe um gene aleatório e substitui seu alelo também de forma aleatória. A troca, por sua vez, escolhe dois genes aleatórios e substitui seus alelos entre si. O operador de mutação nunca é aplicado em alelos com dependências de domínios. Assim como acontece para a taxa de cruzamento, o usuário também pode ajustar a taxa de mutação.

A execução da solução GeSeMa acontece em dois processos principais: (i) validação e aplicação da requisição e configurações genéticas; e (ii) criação e evolução da população. O primeiro processo identifica os arquivos e parâmetros de entrada e os analisa de modo a verificar sua validade. Inicialmente, o arquivo de requisição é validado conforme o modelo especificado na Subseção 3.1, mapeando as estruturas definidas em alto nível para elementos iteráveis pela solução genética (*i.e.*, lista e dicionários padronizados). Em seguida, as configurações genéticas já citadas (*i.e.*, tamanho da população, tamanho do torneio de seleção, operador/taxa de cruzamento e operador/taxa de mutação), em conjunto com o número máximo de gerações (também definido pelo usuário), são verificados e, caso sejam válidos, são utilizados para inicializar os operadores genéticos da solução. A conclusão do primeiro processo gera um conjunto de recursos que serão utilizados para a determinação e evolução de indivíduos aptos a resolver o problema de mapeamento durante o segundo processo.

A execução do segundo processo da solução GeSeMa é sumarizada na Figura 2. Inicialmente, o serviço de rede requisitado, provido na forma de uma cadeia de caracteres

de acordo com a gramática SCAG [Fulber-Garcia et al. 2020], é remodelado para ser processado pelas soluções genéticas (Figura 2: A e B). Além disso, uma população inicial é gerada de forma a criar um conjunto aleatório de indivíduos válidos quanto à topologia da rede (transições entre domínios válidas) e às dependências de domínio (fixando funções com dependências em seus respectivos domínios). Em seguida, é realizada a avaliação dos indivíduos (Figura 2: C) onde estes são verificados quanto à sua validade em relação a restrições de recursos dos domínios escolhidos e políticas. Assim, um procedimento iterativo de agregação define o valor dado a cada uma das métricas de otimização para cada um dos indivíduos avaliados. Então, uma sequência de seleções (Figura 2: D) e de aplicações dos operadores genéticos de cruzamento e mutação (Figura 2: E e F, respectivamente) é realizada a fim de evoluir a população. Todas as etapas ilustradas pela Figura 2: C, D, E e F consistem das operações aplicadas para a conclusão de uma geração (Figura 2: G). A cada geração os melhores resultados (fronteira de Pareto) são salvos separadamente para serem reconsiderados em gerações posteriores. Ao final de um número definido de gerações, a fronteira de Pareto determinada pela solução é retornada como resultado do processamento (Figura 2: H).

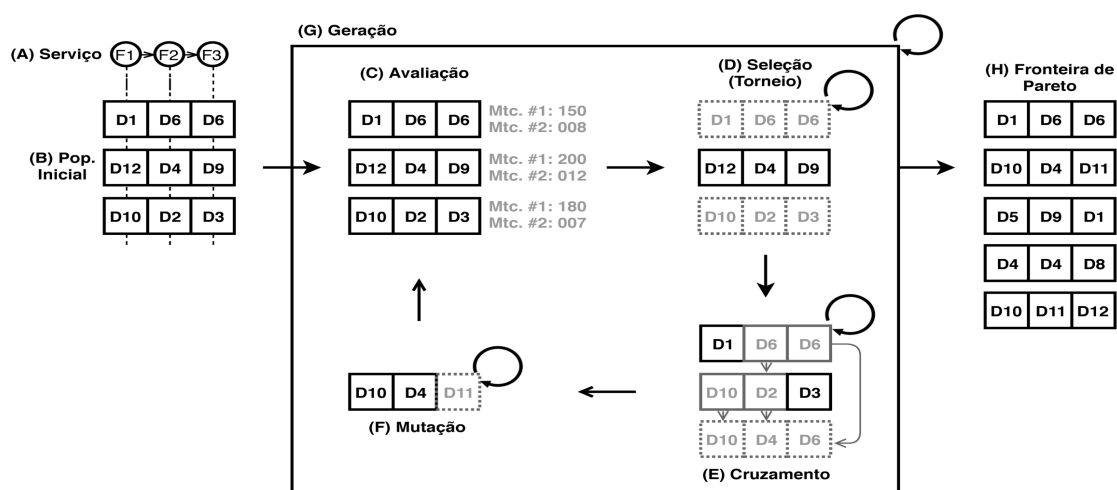


Figura 2. Sumarização do Processamento GeSeMa

Em especial, a etapa de avaliação executa diversas tarefas que viabilizam e otimizam as demais etapas do algoritmo genético. A avaliação é realizada de forma iterativa, percorrendo o cromossoma de cada indivíduo, gene por gene. Assim, a cada iteração são calculadas as métricas locais de acordo com o alelo (domínio) e as métricas de transição, caso necessário, quando um alelo do gene anterior de uma função de rede conectada tem valor diferente do alelo do gene atual. É importante notar que, além do seu próprio alelo, cada gene apresenta um vetor de conexões contendo índices de genes anteriores, estes indicando as funções de rede diretamente conectadas à função de rede representada pelo gene atual. Dessa forma, topologias de rede com ramificações podem ser representadas por um cromossoma linear. O conjunto de avaliações de cada alelo são todos finalmente agregados e então usados para classificar o indivíduo avaliado em relação à dominância de Pareto. Indivíduos considerados inválidos são incluídos em uma lista tabu. Assim, em caso de uma nova ocorrência dos mesmos duas ações podem ser tomadas: substituição do indivíduo por um novo indivíduo aleatório (caso sejam infringidas restrições topológicas e políticas) ou redução de redundância de domínios (caso sejam infringidas restrições de

recursos). Dependências de domínio não são tratadas durante a avaliação, já que nunca são infringidas. Isso ocorre devido à garantia das mesmas na população inicial e à não aplicação de mutações nos alelos dos genes representantes de funções com dependências.

4. Experimentação e Resultados

A proposta foi validada através de um estudo de caso de mapeamento de caches multimídia hierarquizadas em um ambiente distribuído com múltiplos domínios. O serviço de teste apresenta uma topologia linear que conecta sequencialmente n funções de cache multimídia. A primeira função cache é a mestre e quaisquer atualizações e adições por parte do servidor são feitas a partir dela. As demais funções recebem atualizações da sua antecessora, fazendo a disseminação de maneira hierárquica. Cada domínio deve receber, no máximo, uma função cache. O problema de mapeamento consiste de dois objetivos customizados definidos apenas sobre métricas de transição: maximização da capacidade dos canais entre os domínios e maximização da distância geográfica entre caches vizinhas (*i.e.* de hierarquia imediatamente inferior/superior). Dessa forma, o mapeamento deve privilegiar a distribuição geográfica do serviço em zonas de alta largura de banda.

A topologia de rede utilizada consiste de 30 domínios dispostos em um grafo completo, sendo que cada conexão (aresta) entre os domínios (vértices) recebe um valor aleatório v contido em um intervalo fechado $[v_{min}; v_{max}]$ para cada métrica de otimização. Além disso, um conjunto de configurações genéticas foi testado modificando características relacionadas aos operadores e taxa de cruzamento, operadores e taxa de mutação, tamanho do torneio e tamanho da população. Dessa forma, determinou-se a utilização do algoritmo SPEA2, operador de cruzamento SBX com taxa de cruzamento de 30%, operador de mutação de redefinição com taxa de mutação de 5%, seleção por torneio binário e tamanho da população igual a 50 indivíduos. Essa configuração foi adotada pois apresentou a maior média de indivíduos na fronteira de Pareto definida considerando todos os resultados de todas as configurações testadas para um mesmo número de gerações. Os testes apresentados a seguir usam cadeias com 7, 9 e 11 funções de rede e foram executados em uma máquina com processador Intel Core I5-3330 (3.0 GHz) e 8GB RAM (DDR3, 1333 MHz). A quantidade de funções mínima foi definida de forma a criar, junto com a topologia de rede utilizada, problemas intratáveis por soluções exaustivas (*e.g.*, [Riera et al. 2016]) dadas as limitações de memória da máquina utilizada.

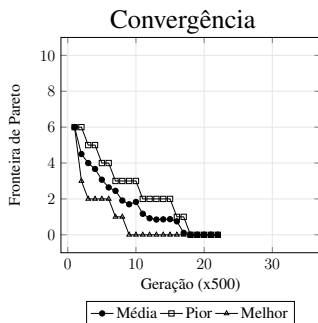


Figura 3. Conv.: 7 Funções

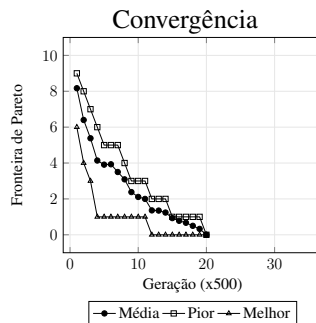


Figura 4. Conv.: 9 Funções

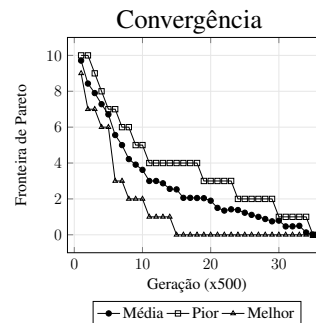


Figura 5. Conv.: 11 Funções

O primeiro experimento realizado consiste do teste de convergência do algoritmo genético. Esse teste verifica a viabilidade da solução GeSeMa para resolver o problema

proposto, definindo se ela é capaz de evoluir e aprimorar os indivíduos ao decorrer das gerações e, eventualmente, convergir para uma fronteira de Pareto (mesmo que não a ótima global). Nesse experimento, as topologias de serviço foram submetidas à execução de um número indeterminado de gerações, sendo o critério de parada a não modificação da fronteira de Pareto após 500 gerações, indicando assim a convergência dos indivíduos na fronteira de Pareto (*i.e.* melhores resultados encontrados pela solução - 0).

As Figuras 3, 4 e 5 apresentam os resultados relacionados à dominância de Pareto (melhor, pior e média da fronteira a qual pertencem os indivíduos gerados em relação aos indivíduos dominantes recuperados ao final do teste) para, respectivamente, as topologias de serviço com 7, 9 e 11 funções. Observa-se que a quantidade de saltos entre fronteiras é diretamente proporcional ao tamanho da topologia de rede submetida ao processo de mapeamento. Esse comportamento decorre do aumento do espaço de busca, possibilitando que um maior número de resultados subótimos seja considerado inicialmente. Entretanto, a quantidade de saltos entre fronteiras são realizados mais rapidamente em um espaço de busca maior uma vez que os mesmos indivíduos subótimos são substituídos mais rapidamente (*i.e.*, em um menor número de gerações) por novos indivíduos dominantes. Finalmente, a quantidade de gerações necessárias para alcançar a convergência do algoritmo também é diretamente proporcional ao tamanho do espaço de busca. Para topologia de serviço com 7 funções, o algoritmo convergiu após 9000 gerações e gastou mais 3500 gerações aprimorando os resultados dentro da própria fronteira de Pareto; já para as topologias de serviço com 9 e 11 funções de rede a convergência ocorreu após a execução de, respectivamente, 10000 e 17500 gerações.

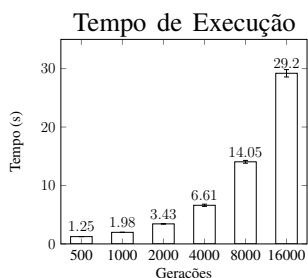


Figura 6. Tempo: 7 Funções

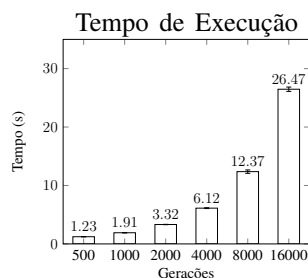


Figura 7. Tempo: 9 Funções

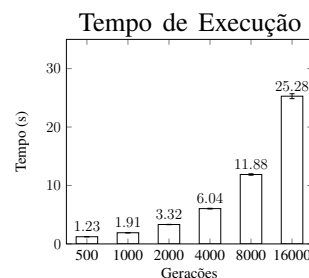


Figura 8. Tempo: 11 Funções

O segundo experimento objetiva verificar a progressão temporal em relação ao número de gerações avaliadas pela solução GeSeMa. Os resultados de tempo de execução (em segundos) para as topologias de serviços com 7, 9 e 11 funções de rede são apresentados, respectivamente, nas Figuras 6, 7 e 8. Todos os experimentos foram repetidos 30 vezes de forma a alcançar um intervalo de confiança de 95%. Através dos resultados obtidos é possível determinar que, para o estudo de caso, a progressão do tempo de execução ocorre de forma linear desde que mantido um mesmo conjunto de configurações genéticas sendo alterada apenas a quantidade de gerações executadas. Note que a complexidade do problema cresce proporcionalmente ao tamanho do espaço de busca – fator que neste estudo de caso varia pelo aumento do número de funções de rede na cadeia a ser mapeada. Entretanto, existe uma relação inversa quando relacionados o tamanho do espaço de busca e o tempo de execução da solução para um mesmo número de gerações. Esse comportamento é consequência da restrição de recursos computacionais aplicada para ga-

rantir a instanciação de, no máximo, uma função cache em cada domínio. Dessa forma, em espaços de busca maiores decorrentes apenas do aumento no número de genes no cromossoma, a probabilidade de criação de um indivíduo inválido devido à reutilização de alelos é maior, aumentando o número de descartes e reduzindo o tempo gasto com avaliações. Entretanto, apesar de reduzir o tempo de execução, esse comportamento dificulta o aprimoramento dos indivíduos e a convergência do algoritmo genético.

Os resultados demonstram que as heurísticas genéticas são adequadas para resolverem problemas de otimização de mapeamento multidomínio de serviços de rede. Apesar de não garantirem o resultado ótimo do mapeamento, as heurísticas podem ser usadas na busca por resultados factíveis e satisfatórios em um tempo adequado à aplicação, sendo estes parâmetros regulados através das configurações genéticas (*e.g.*, tamanho da população e número de gerações). Finalmente, destaca-se também a capacidade de modelagem da solução proposta para comportar e resolver problemas customizados de mapeamento. Assim, através do estudo de caso apresentado, o GeSeMa foi capaz de sugerir mapeamentos candidatos para um cenário complexo e com restrições não resolvível – devido a limitações de recursos dos equipamentos utilizados – por soluções exatas.

5. Conclusão

Um dos principais processos da operacionalização de um serviço de rede NFV consiste da sua implantação no ambiente virtualizado. Esse processo é realizado através de três tarefas principais: composição, integração e programação da execução. Em especial, a técnica de mapeamento multidomínio é utilizada durante a integração quando um serviço deve ser distribuído em múltiplos pontos de presença. Atualmente, a integração de serviços de rede virtualizados é realizada por soluções estáticas e pouco customizáveis em relação aos seus objetivos de otimização e características de avaliação. Dessa forma, este trabalho apresentou o *Genetic Service Mapping*, uma solução flexível que aplica heurísticas genéticas para realizar um mapeamento customizado de topologias de serviço em ambientes multidomínio. Por fim, demonstramos a viabilidade da solução desenvolvida através de resultados de execução da mesma em um estudo de caso específico infactível para soluções exatas nos equipamentos utilizados. Em trabalhos futuros objetiva-se utilizar a solução GeSeMa como um *daemon* que evolui continuamente o mapeamento do serviço, sugerindo integrações e migrações em tempo real ao analisar mudanças na topologia de rede e nas métricas de otimização.

Referências

- Abujoda, A. et al. (2016). Distnse: Distributed network service embedding across multiple providers. In *IEEE Int. Conf. on Communication Systems and Networks*, pages 1–8.
- Boubendir, A. et al. (2016). Naas architecture through sdn-enabled nfv: Network openness towards web communication service providers. In *IEEE/IFIP Network Operations and Management Symposium*, pages 722–726.
- Cao, J. et al. (2016). Vnf placement in hybrid nfv environment: Modeling and genetic algorithms. In *IEEE Int. Conf. on Parallel and Distributed Systems*, pages 769–777.
- Carpio, F. et al. (2017). Vnf placement with replication for load balancing in nfv networks. In *IEEE Int. Conf. on Communications*, pages 1–6.

- Deb, K. et al. (2002). A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE Transactions on Evolutionary Computation*, 6(2):182–197.
- Dietrich, D. et al. (2015). Network service embedding across multiple providers with nestor. In *IFIP Networking*, pages 1–9.
- Even, G. et al. (2019). On-line path computation and function placement in sdns. *Springer Theory of Computing Systems*, 63(2):306–325.
- Fulber-Garcia, V. et al. (2020). CUSCO: A Customizable Solution for NFV Composition. In *Int. Conf. on Advanced Information Networking and Applications*, pages 204–216.
- Garcia, V. F. et al. (2019). Uma solução flexível e personalizável para a composição de cadeias de função de serviço. In *Workshop de Gerência e Operação de Redes e Serviços*, pages 99–112.
- Gil-Herrera, J. et al. (2017). A scalable metaheuristic for service function chain composition. In *IEEE Latin-American Conference on Communications*, pages 1–6.
- Handley, M. (2006). Why the internet only just works. *BT Technology Journal*, 24(3):119–129.
- Hasançebi, O. et al. (2000). Evaluation of crossover techniques in genetic algorithm based optimum structural design. *Computers & Structures*, 78(1-3):435–448.
- Herrera, J. G. et al. (2016). Resource allocation in nfv: A comprehensive survey. *IEEE Transactions on Network and Service Management*, 13(3):518–532.
- Khebbache, S. et al. (2018). A multi-objective non-dominated sorting genetic algorithm for vnf chains placement. In *IEEE Annual Consumer Communications & Networking Conference*, pages 1–4.
- Ma, N. et al. (2017). A model based on genetic algorithm for service chain resource allocation in nfv. In *IEEE Int. Conf. on Computer and Communications*, pages 607–611.
- NFVISG, E. (2012). Network functions virtualization: White paper. Technical report, European Telecommunications Standards Institute.
- Quinn, P. et al. (2015). Problem Statement for Service Function Chaining - RFC 7498. Technical report, Internet Engineering Task Force.
- Riera, J. F. et al. (2016). Tenor: Steps towards an orchestration platform for multi-pop nfv deployment. In *IEEE Conf. on Network Softwarization*, pages 243–250.
- Tavakoli-Someh, S. et al. (2019). Utilization-aware virtual network function placement using nsga-ii evolutionary computing. In *Conf. on Knowledge Based Engineering and Innovation*, pages 510–514.
- Wang, Y. et al. (2017). Cost-efficient virtual network function graph (vnfg) provisioning in multidomain elastic optical networks. *IEEE Journal of Lightwave Technology*, 35(13):2712–2723.
- Zhang, Q. et al. (2016). Vertex-centric computation of service function chains in multi-domain networks. In *IEEE Conf. on Network Softwarization*, pages 211–218.
- Zitzler, E. et al. (2001). Spea2: Improving the strength pareto evolutionary algorithm. *TIK Report*, 103.