Workgroup: SFC
Internet-Draft: draft-wu-sfc-scheduling-implementation-report-00
Published: 23 December 2022
Intended Status: Informational
Expires: 26 June 2023
Authors: X. Wu, *Zhejiang Gongshang University*
C. Hong, *Zhejiang Gongshang University*
C. Li, *Zhejiang Gongshang University*

# Implementation Report for Service Function Chain Scheduling Algorithm

## Abstract

This document provides the application examples of mapping and deployment algorithms to address the problems of large resource overhead and end-to-end latency in Service Function Chain(SFC) that cannot meet the requirements of latency-sensitive applications in terms of both latency optimization and resource optimization.In terms of delay-optimized mapping and deployment of SFC, the application example of granularity-variable SFC mapping algorithm based on microservice architecture reduces the number of instantiated microservice units and the number of end-to-end routing hops by merging redundant microservice units within the service function chain and reusing microservice units between the service function chains.In terms of resource-optimized mapping and deployment of SFC, the application example of SFC mapping algorithm based on improved gray wolf optimization algorithm optimizes the mapping of SFC through two strategies of reverse learning initialization and nonlinear convergence factor, and improves the efficiency of the mapping scheme.

## Status of This Memo

## Copyright Notice

## Table of Contents

## 1. Introduction

In Network Function Virtualization (NFV) environment, user's service is done by Service Function Chain (SFC) composed by Virtual Network Function (VNF) in a certain order, so user's request can be called SFC Request ( SFC Request, SFCR). Given a series of SFCRs, it requires instantiating and deploying the corresponding VNFs to complete these services. The software features of VNFs allow more flexibility in the selection of deployment locations and resource allocation. In addition, during the process of SFCR mapping, it is necessary to consider the sequential constraints of the VNFs that make up its SFCs to avoid additional bandwidth usage as much as possible. Thus, in the process of SFC scheduling and VNF deployment, it is necessary to design an optimal solution to improve the resource utilization of the network, which is also known as the SFC scheduling and VNF deployment problem.

The scheduling of service function chains and deployment of virtual network functions include three main parts: determination of the mapping scheme, deployment and management, among which the determination of the mapping scheme as a key step in the scheduling will affect the costs associated with the subsequent processes. How to reasonably deploy each virtual network function which constitutes the service function chain in the physical topology and reduce the overhead of physical resources is a major challenge for the current mapping scheme determination problem. In addition, with the development of emerging technologies such as 5G, many latency-sensitive applications such as virtual reality, augmented reality, and autonomous driving have emerged, which put forward high requirements on the end-to-end latency of Service Function Chain, and how to reduce the end-to-end latency of Service Function Chain to a certain extent is another major optimization goal and challenge of the mapping problem.

## 2. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT","SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119] RFC 8174 [RFC8174].

This document uses the following terms from RFC 7665 [RFC7665] RFC 8763 [RFC8763]:

- Network Function Virtualization (NFV)
- Virtualize Network Function (VNF)
- Service Function (SF)
- Service Function Chain (SFC)
- Service Function Chains (SFCs)
- Service Function Chain Request (SFCR)

This document introduces the following terms:

- Grey Wolf Optimizer (GWO)
- Improved Grey Wolf Optimization (IMGWO):Improved standard gray wolf optimization algorithm using a reverse learning-based population initialization strategy and a convergence factor nonlinear optimization strategy.
- IMGWO-SFCM:This document proposes a service function chain mapping method based on the improved GWO algorithm for the service function chain mapping and deployment resource optimization problems.
- VG-SFCM:This document proposes a granular variable service function chain mapping algorithm based on microservice architecture for service function chain mapping and deployment latency optimization problems.
- AtomNF:It is defined as a fine-grained microservice unit that represents the smallest logical unit that operates on data after the virtual network function has been split in the service function chain.

## 3. Example of SFC scheduling implementation algorithm

Network resources in network topology are limited, and Network Function Virtualization (NFV) presents new challenges in terms of how to efficiently map and deploy Service Function Chain (SFC) based on user's Service Function Chain Request (SFCR) while saving operators' cost. This document presents examples of Variable Granularity Service Function Chain Mapping(VG-SFCM) based on microservices architecture algorithm and Improved Grey Wolf Optimization Service Function Chain Mapping ( IMGWO-SFCM ) algorithm for both latency optimization and resource optimization respectively.

### 3.1. Mapping algorithm of granularity variable SFC based on microservice architecture

There are many logical functions that are repeatedly performed between different individual VNFs within a single SFC, such as packet input and output, message classification, etc. In the process of SFC mapping, considering the minimum granularity of the mapping and performing microservice splitting and merging can effectively remove these redundant logical functions and reduce the end-to-end delay and resource overhead incurred during deployment according to the mapping solution. Considering the splitting and optimization of VNFs within SFCs, we propose a Variable Granularity Service Function Chain Mapping algorithm (VG-SFCM) based on microservice architecture.

Firstly, the set of microservice units (AtomNFs) constituting the original SFC is obtained by fast splitting based on typical VNF, and then the redundant AtomNFs within a single SFC are merged by applying the merging strategy of redundant AtomNFs and drawn to form a new SFC service graph, then finally, the instantiation of AtomNFs is further reduced by the AtomNF reuse strategy among SFCs.

The splitting of VNFs is mainly based on the set of VNFs and their corresponding logical functions, and the VNFs are split into several fine-grained AtomNFs, which cannot be fully automated at this stage due to the different functions provided by different VNFs, and still requires a lot of manual work to complete the logical analysis and AtomNF definition. In order to realize the fast splitting of SFC and reduce the resource overhead caused by the splitting, the fast splitting based on typical VNF prepares the logical analysis and splitting work in advance by logically analyzing the VNFs that appear more frequently and preparing the AtomNF image file corresponding to the high-frequency VNFs in advance and putting them into the repository. When a SFCR arrives, it can quickly find and directly pull the relevant mirror according to the relevant VNF splitting scheme. If there is a VNF in SFC that is not pre-split, we do not split it and treat it as a coarse-grained single unit.

Different VNF may get several identical AtomNF after splitting, and repeated execution of the same AtomNF does not bring actual changes to the data flow instead it increases the resource overhead and end-to-end delay of the corresponding mapping nodes, so we can analyze and merge these multi-occurring AtomNF. Referring to the classification rules of MicroNF, AtomNF is classified into six categories: endpoint, classification, modification, reconfiguration, flow control and static for merging. In order to ensure that the new SFC after merging can provide the same services as before merging, while minimizing the end-to-end latency, the merging of AtomNF needs to comply with the following principles.

- The packets must undergo the same processing steps as the original SFC, but some of the redundant steps may be performed less frequently.
- Each AtomNF with multiple executions is subject to a merge likelihood analysis that seeks to compress the end-to-end length of the SFC as much as possible.
- The location of an AtomNF in the new SFC can be changed, but the change cannot cause the data flow to change compared to the original SFC.

The reuse of VNF or AtomNF instances can currently be realized by multi-tenancy technology, which enables a single software instance to provide corresponding services for multiple tenants. Compared with the traditional non-reusable mapping scheme, the node resource utilization can be effectively improved by providing services for SFCRs through one reusable AtomNF instance in SFC mapping process. In order to realize the reuse of subsequent requests, the first arriving SFCR need to determine the best mapping scheme through IMGWO-SFCM. After the subsequent arriving SFCRs complete the initial scheduling and splitting and merging work, they judge the intersection of their AtomNF sets with the AtomNF sets of the previous SFCR, and the resulting intersection is the reusable AtomNFs, and then perform the reusability judgment of the relevant microservice unit mapping nodes.After the screening of reusable AtomNFs is completed, the corresponding node is checked for resources to determine whether the remaining resources of the node can carry the remaining non-reusable AtomNFs of the SFC. If it cannot, the relevant AtomNFs are deployed to the neighboring nodes with sufficient resources. When there is no available resource in the neighboring nodes to meet the relevant demand, the mapping scheme is determined by IMGWO-SFCM for the SFC again.

Using splitting, merging and multiplexing strategies for SFC, VG-SFCM reduces the instantiation of AtomNF, and effectively reduces ends-to-ends latency of SFC, which can also minimize the cost of network deployment to a certain extent.

## 3.2. Improved Grey Wolf Optimization Service Function Chain Mapping

For SFCR initiated by users at the application layer, the control layer needs to orchestrate it into SFC after receiving it. The SFC deployed according to different mapping schemes consume different computing resources, memory resources and bandwidth resources, and also incur different node activation costs and deployment costs.

The mapping and deployment of SFC needs to obey the following four basic principles.

- The end-to-end latency of SFC after deployment needs to be within the maximum acceptable latency of user requests.
- The computing resources provided by the physical node selected for SFC after deployment need to be greater than what is required by all VNFs deployed in that node.
- The bandwidth of the link through which the data flow between the physical nodes selected by SFC needs to meet the bandwidth requirements of SFC.

- The direction of the data flow needs to strictly follow the order of SFC constructed for user requests.

First, we search the first K paths between the source and destination nodes by the loop-free KSP search algorithm and search the possible mapping schemes according to the first K shortest paths and the number of VNFs to be deployed in the SFC. Subsequently, the matching of individual gray wolves with the mapping scheme is completed by mapping scheme coding. Then, we initialize the gray wolf population according to the backward learning strategy of Improved Gray Wolf Optimization algorithm (IMGWO), select the dominant gray wolf individuals according to the optimization function, i.e., the fitness function, and then enhance the global search and local search ability of the algorithm in the early stage by the nonlinear convergence strategy of the convergence factor. Finally, we determine the final mapping scheme by iteration.

The standard gray wolf optimization algorithm suffers from slow late convergence and easily falls into local optimum, so an improved gray wolf optimization algorithm (IMGWO) is proposed by improving the standard gray wolf optimization algorithm through the group initialization strategy based on reverse learning and the convergence factor nonlinear optimization strategy. The gray wolf optimization group initialization based on the backward learning strategy first randomly generates the initial wolf group and generates the reverse wolf group based on the initial wolf group. Subsequently, the top three individuals with the highest fitness in the two groups are determined according to the fitness function. Meanwhile, in order to improve the global search ability and local search ability of the standard GWO algorithm and avoid falling into local optimal solutions, the nonlinear optimization strategy is applied to the convergence factor a.

The group initialization strategy based on backward learning and the convergence factor nonlinear optimization strategy make IMGWO algorithm a good balance of global search and local search ability, which can accelerate the overall convergence speed of the algorithm.

The IMGWO algorithm can be divided into seven steps as follows.

1. Set the initialization size of gray wolf population, the maximum number of iterations, initialization and other parameters.
2. Generate the initial group based on the results of random initialization using reverse learning.
3. Calculate the fitness value of each individual gray wolf in the group, select the top three dominant wolves according to the fitness value, and record their positions.
4. Update the value of convergence factor according to the convergence factor nonlinear optimization strategy and update the values of each parameter.
5. Update the individual positions of gray wolves.
6. Calculate the group fitness value after updating the individual positions of gray wolves and update the dominant wolves and their positions.
7. Judge whether the constraint of the algorithm is reached, or the maximum number of iterations is reached, if it is met, the algorithm ends and the optimal solution is output; if it is not met, return and re-execute steps 3 to 6.

The steps of the implementation of IMGWO-SFCM algorithm are as follows.

1. First, the physical node resources, computing resources and bandwidth resources required for SFC are calculated based on the network topology and SFC service diagram, and whether a single-node deployment can be performed based on the corresponding resource status. If the node has sufficient resources for SFC deployment, it further determines whether the end-to-end delay after deployment can meet the lowest delay requirement of users. For the case where multiple nodes meet the lowest latency requirement for individual deployment, the physical nodes are mapped according to the principle of minimizing the end-to-end delay of SFC.
2. After finishing the judgment of single-node mapping possibility, the algorithm starts to perform the steps related to the determination of multi-node mapping scheme. Firstly, according to the source and destination nodes of SFC, it applies the qualified loop-free KSP algorithm to perform the screening of the first K shortest paths, and then it performs further search of possible mapping schemes through the single-link mapping scheme, which enables the initialization range of gray wolf group to be determined. Finally, the best mapping scheme is output through the IMGWO related process.

## 4. Acknowledgements

Thanks to Hong Chen and Zhang Junnan for the first draft.

## 5. Normative References

**[RFC2119]** Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997, <http://xml.resource.org/public/rfc/html/rfc2119.html>.

**[RFC7665]** Halpern, J., Ed. and C. Pignataro, Ed., "Service Function Chaining (SFC) Architecture", RFC 7665, DOI 10.17487/RFC7665, October 2015, <https://www.rfc-editor.org/info/rfc7665>.

**[RFC8174]** Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

**[RFC8763]** Rahman, A., Trossen, D., Kutscher, D., and R. Ravindran, "Deployment Considerations for Information-Centric Networking (ICN)", RFC 8763, DOI 10.17487/RFC8763, April 2020, <https://www.rfc-editor.org/info/rfc8763>.

# Authors' Addresses

**Xiaochun Wu**
Zhejiang Gongshang University
18 Xuezheng Str., Xiasha University Town
Hangzhou
Phone: +86 13067732553
Email: spring-403@zjgsu.edu.cn

**Chen Hong**
Zhejiang Gongshang University
18 Xuezheng Str., Xiasha University Town
Hangzhou
Phone: +86 19157703315
Email: 20020090025@pop.zjgsu.edu.cn

**Chuanhuang Li**
Zhejiang Gongshang University
18 Xuezheng Str., Xiasha University Town
Hangzhou
Phone: +86 571 28877723
Email: chuanhuang_li@zjsu.edu.cn