A Hybrid Flow/Packet Level Model for Predictive Service Function Chain Selection in SDN

Frank Wetzels[1(✉)], Hans van den Berg[2], and Rob van der Mei[1]

[1] CWI, Amsterdam, The Netherlands
{f.p.m.wetzels,r.d.van.der.mei}@cwi.nl
[2] TNO, The Hague, The Netherlands
j.l.vandenberg@tno.nl

Abstract. This paper is motivated by recent developments in SDN and NFV whereby service functions, distributed over a centralised controlled network, are connected to form a service function chain (SFC). Upon arrival of a new service request a decision has to be made to which one of SFCs the request must be routed. This decision is based on (1) actual state information about the background traffic through the SFC nodes, and (2) a prediction of the fraction of time that the SFC is in overflow during the course of the new flow in the system. In this paper, we propose a new method for assigning an incoming flow to an SFC. For that, we propose and compare two methods: a simple flow-based algorithm and a more refined hybrid flow/packet-based algorithm. By extensive simulations, we show that the simple flow-based algorithm works particularly well if the network is not overloaded upon new flow arrival. Moreover, the results show that the flow/packet-based algorithm enhances the flow-based algorithm as it handles initial overload significantly better. We conclude that the prediction-based SFC selection is a powerful method to meet QoS requirements in a software defined network with varying background traffic.

Keywords: Software defined network · Service function chain · Predictive selection · Varying background traffic

1 Introduction

In modern networks, the decoupling of (network) functions from the underlying hardware and the decoupling of the control plane from the data plane in network devices, has drawn a lot of attention. This led to the development of the network function virtualisation infrastructure (NFVI) [1] and software defined networking (SDN) [2] respectively. SDN, the central controlled network, makes it possible to steer traffic to functions anywhere located in the network [3]. These functions can be of any type of operation applied to traffic. For example, internet traffic

Q. Zhao and L. Xia (Eds.): VALUETOOLS 2021, LNICST 404, pp. 93–106, 2021.
https://doi.org/10.1007/978-3-030-92511-6_6

flowing into the network that needs to be scanned for viruses, intrusion detection and spyware. Functions, to be applied sequentially to traffic that passes are called a service function chain (SFC) [4]. The flexibility of SDN and NFVI leads to SFC resource allocation (SFC-RA) identified by the following stages: Chain composition, embedding and scheduling. As per [5–7], for each stage an optimal or near optimal solution can be determined by using specific algorithms to which stage specific conditions apply to process service requests (SRs) by the network. Different types of algorithms have been proposed to optimise the concatenation of VNFs with fixed capacity to provide a chain composition. At the embedding stage most algorithms take the chain composition as input and determine the location of where the VNFs need to be running in the network given a set of requirements like CPU and memory use. Scheduling applies to a set of different operations individually used in different, possibly multiple, chains. The sequence of required operations for each chain and the availability of resources lead to an optimal allocation sequence in time for each operation per chain.

In this paper we take a different approach by focussing on the background (BG) traffic characteristics for SFC selection. Figure 1 gives an example whereby an SR enters the network, indicated by the blue straight arrow. We assume the presence of SFCs at fixed locations in the network each consisting of identical functions applied in the same sequence and one type of SR entering the network. Two SFCs are shown and a choice must be made to which SFC the SR needs to be directed to upon arrival. In particular, we assume the presence of background traffic (indicated by the red dashed arrows) that is handled by the individual SFC nodes as well, but is not part of the SR. The background traffic affects the resource availability on each individual SFC node. We will assume certain characteristics of the background traffic and use this knowledge to make a predictive selection of the SFC that affects the SR the least. The SR is then steered to the SFC of choice until it finishes. Throughout, we will call the SR a foreground (FG) flow and the other flows BG flows.

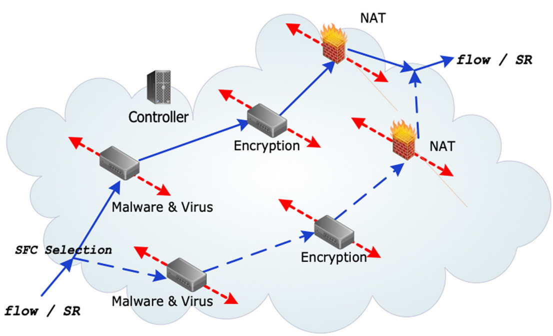

Fig. 1. Flow to be steered to one SFC based on BG traffic violating the critical level. (Color figure online)

The contribution of this paper is three-fold: We show a model and method to use characteristics of the background traffic that flows through individual SFC nodes to make a predictive SFC selection. In addition, we propose two algorithms: One that uses flow characteristics only to make a predictive SFC selection and a second algorithm that adds background packet behaviour in addition to flow characteristics to enhance the first algorithm. The second algorithm's enhancement is clearly noticed in case of high intense background traffic upon arrival of the SR. To the best of the authors' knowledge, this has not been considered before.

2 Problem Description

The problem in deciding which SFC to choose for a newly incoming FG-flow is caused by the inherent *uncertainty* about the number BG flows that travel through SFC nodes during the course of the FG-flow in the system. BG flows travelling through SFC nodes affect the resource availability dynamically, which in turn affects the processing of the FG flow travelling through the SFC itself. If the BG flows are too intense at an SFC node, available resources may be too low to process a FG flow successfully, given QoS requirements.

To effectively deal with this uncertainty, the challenge is to determine at each SFC node the expected amount of time the number of BG flows are at - or below - some level such that during the lifetime of the FG flow, the processing of the FG flow is not affected long enough by the BG flow intensity. Note that the term 'not long enough' is dictated by the requirements to which the FG flow should comply to.

To determine the expected violation duration, we need to define a critical level at each SFC node such that if BG flows do not exceed this level too long, the available resources are sufficient to process the FG flow while fulfilling FG flow requirements. As a result, we can determine what SFC affects the FG flow the least at individual SFC nodes. Secondly, at each SFC node, we will assume the BG flows are driven by a birth-death (BD) process [8]. This assumption enables us to use results of performance evaluation on BD processes in [9]. To illustrate this, Fig. 2 gives a realization of the state of the BD process (represented by X_t) over time. The red dashed horizontal line represents the critical level, $m = 6$. During the lifetime of the FG flow, the number of BG flows exceeds this threshold a certain amount of time.

In [9] a method is presented that enables us to calculate the expected amount of time a BD process is above a critical level during the lifetime of the FG flow. The expected violation duration enables us to predict which SFC affects the processing of the FG flow the least. However, the computational afford is intense. To circumvent this problem, we will use pre-calculated tables for fast decision making. In [10], a pre-calculated table is used for determining whether an SLA is met for a high-priority flow using the method in [9]. We will follow a similar method and create tables beforehand and use these for fast lookup to determine the SFC of choice. The above method ([9]) will be applied in our first algorithm. We will refer to this algorithm as a flow-based (FB) algorithm.

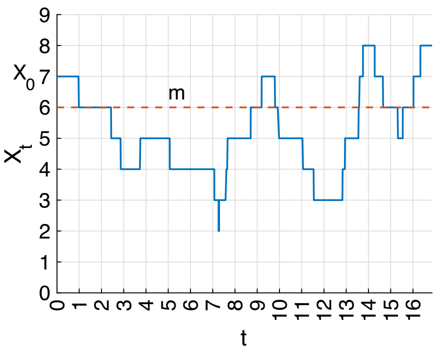

Fig. 2. Time variation in the number of BG flows. (Color figure online)

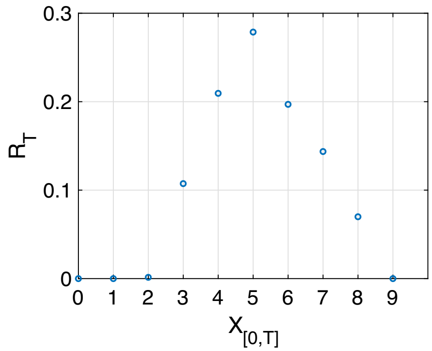

Fig. 3. Expected fractional BG flow presence duration R_T.

To enhance the FB algorithm, we will consider a second algorithm which takes into account the packet intensity per individual flow as well. This algorithm is called a flow/packet-based (FPB) algorithm. For that we will make use of the method presented in [11]. It enables us to include the transient mean sojourn time BG flow packets have when entering the SFC node.

We will combine the knowledge of BG flows above individual levels to determine the fractional presence duration for each number of BG flows. In Fig. 3 an example is given. For each number of $X_{[0,T]}$ BG flows, its expected fractional presence duration R_T during the life time $[0, T]$ of the FG flow is given. Combining the packet transient mean sojourn time together with the fractional presence duration of BG flows, a weighted transient BG packet delay can be determined. Lookup tables will be created beforehand, used for calculation and fast decision making.

3 Model and Method for Selective SFC Choice

Section 3.1 describes the network and context assumed in this paper. Next, in Sect. 3.2, additional network details, assumptions and definitions are provided. The two service chain selection algorithms, FB and FPB, will then be described in Sects. 3.3 and 3.4 respectively. Details of the analyses needed to run these algorithms are provided in Sect. 4.

3.1 Network Definition

In Fig. 4, a network is given, consisting of an entrance node `A`, an exit node `B`, an SDN controller and C parallel chains of N nodes in length. The FG flow enters the network at node `A` where it is directed (routed) to the chain that is expected to provide the best performance. The FG flow, undergoes the operations applied by the nodes and leaves the network at node `B`. All chains apply

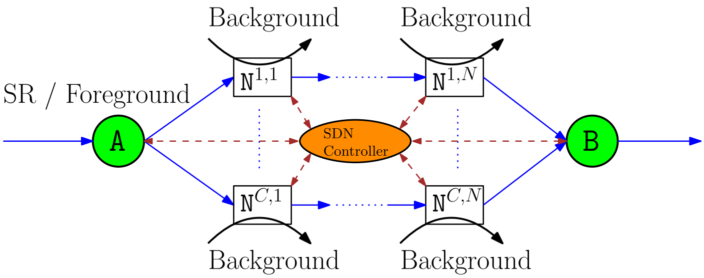

Fig. 4. C SFCs with BG flows each consisting of N-nodes.

the same functions in the same sequence. At each SFC node, BG flows enter the SFC node, undergo the same operations as the FG flow and leave the node. There is no dependency between BG flows at different SFC nodes. In this study we will focus on a single FG flow travelling through the whole chain. The number of BG flows present at a node is driven by a BD process; the BD processes for the different nodes are independent. The routing decision at A is made by the SDN controller.

3.2 Assumptions and Definitions

Numerous parameters can be varied. For example, the birth and death rates of BG flows at the various nodes, the inter-arrival time of packets per one flow of BG and FG flows, etc.

The communication between the SDN controller and the nodes, the transmission speeds, packets lengths and delay on the links are not considered. No packet loss occurs. BG flows arrive at an SFC node, leave the node after processing and will not be considered thereafter.

The following general definitions are made (note the use of the suffices 'f' and 'p' associating to flows and packets respectively):

- **Nodes $\mathtt{N}^{i,j}$:** There are C SF chains, each consisting of N nodes. For each chain $i = 1, \ldots, C$, the nodes are connected in sequence, i.e. the j-th node in chain i, $\mathtt{N}^{i,j}$, accepts packets belonging to the FG flow from $\mathtt{N}^{i,j-1}$ and forwards these packets to $\mathtt{N}^{i,j+1}$ for $j = 2, \ldots, N-1$. Packets to $\mathtt{N}^{i,1}$ come from A. Packets to B come from $\mathtt{N}^{i,N}$.
- **BG BD process $\{X_t^{i,j} | t \geq 0\}$:** New BG flows arrive at node $\mathtt{N}^{i,j}$ according to a Poisson process with rate $\lambda_f^{i,j}$, and have an exponentially distributed lifetime (duration) with parameter $1/\mu_f^{i,j}$. $X_t^{i,j}$ represents the number of BG flows present at $\mathtt{N}^{i,j}$ at time t. The BG flow arrival and departure processes at the different nodes are independent. The state-space $S_M^{i,j}$ of the BG BD process associated to node $\mathtt{N}^{i,j}$ may be limited by a certain maximum $M^{i,j}$.
- **Critical level $m^{i,j}$:** The critical level of BG flows at node $\mathtt{N}^{i,j}$ (i.e. the number of BG flows at which the node gets overloaded) is represented by $m^{i,j}$.

- **BG packets rates $\lambda_p^{i,j}$:** Each BG flow consists of packets arriving at an SFC node according to a Poisson process at rate $\lambda_p^{i,j}$.
- **FG flow duration T and packet rates $\hat{\lambda}_p$:** We consider one FG flow of an exponential distributed duration T with parameter τ. The FG flow consists of packets arriving according to a Poisson process at rate $\hat{\lambda}_p$.
- **SFC nodes service rates $\mu_p^{i,j}$:** The Nodes $\mathtt{N}^{i,j}$ process packets with service rate $\mu_p^{i,j}$.
- **SFC queue:** At all nodes, all packets enter an infinitely large FIFO queue. FG and BG packets are treated the same way.

3.3 Flow Based Decision Algorithm

The network is running for a while and at some point in time a FG flow arrives at node A. This is the moment $t = 0$s. The FG flow's duration is T, exponentially distributed with parameter τ. At that moment the number of BG flows arriving at all SFC nodes is recorded. That is, at node j in chain i, $X_0^{i,j}$ is set to the number of BG flows. We will leave out the indices i and j for readability hereafter.

The FB algorithm applies the method in [9] to each SFC node, whereby the BG flows are driven by BD process $\{X_t | t \geq 0\}$ with BD parameters λ_f and μ_f respectively. This gives us $E_{T,m,n}$, the fraction of the expected amount of time X_t is above critical level m at an SFC node during $t \in [0, T]$ with n BG flows at $t = 0$.

Each chain consists of a sequence of nodes. The affect of each chain to the FG flow can be determined. The chain that affects the FG flow the least will become the chain of choice to process the FG flow.

Note that the assumption on the flow characteristics may seem somewhat restrictive. The results in [9] may need to be extended to non-exponential distributions, as mentioned in Sect. 6.

3.4 Flow/Packet-Based Decision Algorithm

The same prerequisites apply as with the flow-based algorithm. However, the chain of choice is selected differently. Per SFC node, the FBP algorithm combines two parameters to create a weighted expected packet delay: The fraction of expected amount of time of the *presence* of k BG flows ($R_{T,k,n}$), for all k in the state-space S_M, and the transient mean sojourn time ($\mathtt{TMST}_k$) of packets belonging to the presence of k BG flows.

Presence Duration of BG Flow

Determine $E_{T,k,n}$, the fraction of expected amount of time BG flows are above some level k, for all $k \in S$ during the lifetime of the FG. The number of BG flows at $t = 0$ is n. The fraction of expected amount of time of the presence of k BG flows, $R_{T,k,n}$ can be determined from $E_{T,k,n}$ and $E_{T,k+1,n}$.

Transient Mean Packet Delay
By assuming that all k BG flows are of the same type, the packet arrival rate of k BG flows equals to k times the packet rate of one single BG flow. This way we can determine the expected transient packet delay during the presence of k BG flows.

Weigthed Expected Packet Delay
By combining $R_{T,k,n}$ with $\texttt{TMST}_k$ for $k > m, k \in S$, a weighted average on the expected packet delay above m can be determined. For each individual SFC node, the weighted expected transient BG packet delay is determined. The affect of each chain to the FG flow can be determined. The chain that affects the FG flow the least will become the chain of choice to process the FG flow.

The analyses and computations needed to run the FB and FBP algorithms described above will be provided in the next section.

4 Analysis

In this section we present the theory needed to derive the quantities used in the two decision algorithms described in the previous section. We will start, in Sect. 4.1, with the analysis of the behavior of the BD process representing the number of BG flows. In particular we will use the results of [9] to predict the time that the number of BG flows at a node is above a certain critical level during transmission of the FG flow, given the number of BG flows present at the start of its transmission. This analysis is needed for both decision algorithms. Next, in Sect. 4.2, the packet level delay analysis needed for the flow/packet-based decision algorithm is performed. Finally, using these results, the two decision algorithms are further detailed and concertized in Sects. 4.3 and 4.4 respectively.

4.1 Expected Fraction of time Above Critical Level

In this section we present the theory needed for the decision algorithms described in Sect. 3. In particular, we need to determine the fraction of time that the number of BG flows at a node is above a certain critical level during the lifetime (transmission time) of the FG flow, given the number of BG flows present upon the start of the FG flow transmission. Let's therefore focus on one single node, denoted $\texttt{N}$, with the number of BG flows varying according to BD process $\{X_t|t \geq 0\}$, thus leaving out the indexing i, j in order to simplify notation. Define $E_{T,m,n}$ as the expected duration the number of ongoing BG flows is above level m during the lifetime T of the FG flow given the number of ongoing BG flows upon start of the FG flow is n, i.e.

$$E_{T,m,n} := \sum_{r=0}^{M} \mathbb{E}(U_m, X_T = r | X_0 = n).$$

In [9] a method is developed which provides a method for determining $E_{T,m,n}$ in terms of T, m, n and the parameters of the BD process driving the BG flows at the node for a finite state space of size M. This method is concerned with solving a linear system in the Laplace domain. The solutions to this system are Laplace transforms representing the amount of time a BD process is above m, starting at state n at $t = 0$ and ending at state l at $t = T$. Taking the limit to zero of all differentiated transforms of the solutions results in the expected time a BD process is above m, starting in n and ending at l. Finally, $E_{T,m,n}$ is determined by adding up the results for all end states while starting in n. The above method depends on τ, not on T.

With the expected *time* the number of BG flows is above the critical level, we can easily determine the expected *fraction* of time $E^*_{T,m,n}$ it is above m during the lifetime T of the FG flow given n BG flows at the start of its transmission:

$$E^*_{T,m,n} = \frac{1}{T} E_{T,m,n}. \tag{1}$$

Hence, the expected fraction of time that the number of BG flows $\{X_t | t \geq 0\}$ stays at or is below m during $[0, T]$ is given by $1 - E^*_{T,m,n}$. The above results (1) can be applied to each SFC node $\mathtt{N}^{i,j}$. In particular, let $E^{i,j}_{T,m^{i,j},n^{i,j}}$ be representing the expected fraction of time the number of BG flows at node $\mathtt{N}^{i,j}$ is above $m^{i,j}$ starting with $n^{i,j}$ BG flows at the start of the transmission of the FG flow. Then, the expected fraction of time the number of BG flows at *any* node in chain i exceeds the critical level of its associated node, is given by,

$$E^i_T = 1 - \prod_{j=1}^{N} (1 - \frac{1}{T} E^{i,j}_{T,m^{i,j},n^{i,j}}).$$

4.2 Weighted Expected Transient Packet Delay

For the flow/packet-based decision algorithm we are also interested in the expected fraction of time a specific number k BG flows is present during the lifetime of the FG flow. Obviously, the expected fraction of time of the presence of $k \in S_M$ BG flows at a 'general' $\mathtt{N}$, denoted by $R_{T,k,n}$, is given by the expected fraction of time above $k - 1$ minus the expected fraction of time above k, i.e.

$$R_{T,k,n} = E_{T,k-1,n} - E_{T,k,n}, \qquad k = 0, 1, \ldots, M. \tag{2}$$

with $E_{T,k,n} := 0$ for $k, n \geq M$ and $E_{T,-1,n} := T$. Note that $E_{T,k-1,n} \geq E_{T,k,n}$ for $k \in S_M$.

Again, consider one node. At $t = 0$s, BG packets arrive at rate $n\lambda_p$ at node $\mathtt{N}$. During the FG flow presence, the number of BG flows vary. By using [11], the transient mean sojourn time $\mathtt{TMST}_k$ can be determined for the $k\lambda_p + 1$-th BG packet, part of k BG flows, arriving at node $\mathtt{N}$.

With (2), we determine the weigthed expected transient BG packet delay, R^*, for the BG flows above m belonging to node $\mathtt{N}$,

$$R^* = \sum_{k=m+1}^{M} \mathtt{TMST}_k R_{T,k,n}.$$

Apply the above to $\mathtt{N}^{i,j}$. Then, the weighted expected transient delay for chain i, R_T^i, during the lifetime of the FG flow is approximated (see Sect. 4.4) by,

$$R_T^i = \sum_{j=1}^{N} \sum_{k=m^{i,j}+1}^{M^{i,j}} \mathtt{TMST}_k^{i,j} R_{T,k,n^{i,j}}^{i,j}.$$

4.3 Flow-Based Decision Algorithm

The selected chain, is the one with lowest weighted expected fractional duration E,

$$E := \min_{i=1,\ldots,C} E^i.$$

The chain of choice selected by the FB algorithm is given by,

$$COC_{FB} = \{i \in \{1, \ldots, C\} | E_T^i = E\}. \tag{3}$$

4.4 Flow/Packet-Based Decision Algorithm

In [11] a method is presented to calculate the expected transient sojourn time of the r-th packet while w packets are in an M/M/s queue and l packets left the queue, start counting at packet 0. To simplify the calculations, we will determine the expected delay for the $n^{i,j}\lambda_p^{i,j} + 1$-th packet, assuming $n^{i,j}\lambda_p^{i,j}$ packets are in the queue at $t = 0$. The '+1' assumes the next packet is the first packet of the FG flow. Define R as the minimum of expected transient delay of all chains,

$$R := \min_{i=1,\ldots,C} R_T^i.$$

Then the chain of choice will be,

$$COC_{FPB} = \{i \in \{1, \ldots, C\} | R_T^i = R\}. \tag{4}$$

5 Numerical Results

All simulations were conducted on a 12-core CPU system running MATLAB in combination with C-programmed code to speed up the calculation of the transient packet behaviour. To compare the delay of the selected chain against other chains, the FG packets are send to all chains immediately after chain selection. There is no cross-chain influence.

For the FPB algorithm, we combine packet behaviour and critical levels to determine the expected presence duration of the number of BG flows above the critical level. The FB and FPB algorithms have common parameters that will not be changed during simulations and other parameters will be changed during simulations. These are given in Sect. 5.1. In Sect. 5.2, the performance parameters are defined that are used to compare the simulation results in Sect. 5.3.

5.1 Simulation Parameters

The following parameters have proven to be realistic in our case. The nodes process packets at a rate of 120 pps, packets arrive at a rate of 10 pps for all flows. We run 1000 simulations per set of parameters and measure the average delay on all chains for the FG packets. Simulations are run for the FB and FPB algorithms, each with their set of parameters. We choose two chains each consisting of one node for the FB algorithm and two chains each consisting of two nodes for the FPB algorithm. The FG flow duration T is determined per single simulation and has no influence on the SFC selection.

The fixed parameters in all simulations for both algorithms are given in Sect. 5.1. In Sect. 5.1 the hierarchy of the varying parameters is shown.

Common Fixed Simulation Parameters

The common parameters for both algorithms are $(i = 1, \ldots, C, j = 1, \ldots, N)$

- The duration of the FG flow (T) is exponential distributed with parameter $\tau = \frac{1}{10}$.
- The individual BG flow mortality rate $\mu_f^{i,j}$ for all nodes is set to $\frac{1}{5}$.
- The FG packet arrival rate is set to $\hat{\lambda}_p = 10$.
- The BG packet arrival rate is set to $\lambda_p^{i,j} = 10$.
- The service rate at all nodes $\mu_p^{i,j}$ is set to 120.
- The state space of all BD processes is the same: $S_M^{i,j} = 0, 1, \ldots, M$, with $M = 26$, based on [10].

Varying Parameters for FB and FPB Algorithms

By means of Pseudocode 1 an illustration is given of the parameters that are varied and in what hierarchy during the simulations. Note that in line 1 and 3, $n^{i,j}$ and $\lambda^{i,j}$ are set for all i, j respectively. In line 4 and 5, $\lambda_f^{1,N}$ and $n^{C,N}$ are overwritten.

1: **for all** $i, j : n^{i,j} = 5, \ldots, 13$ **do**
2:     **for all** $i, j : m^{i,j} = 5, \ldots, 13$ **do**
3:         $\lambda_f^{i,j} = 1.0, \forall_{i,j}$
4:         $\lambda_f^{1,N} = 2.0$
5:         **for** $n^{C,N} = 0, \ldots, 22$ **do**
6:             1000 Simulations
7:         **end for**
8:     **end for**
9: **end for**

PseudoCode 1: Parameter variation and hierarchy.

5.2 Performance Metric

To compare the algorithms we need to define a performance metric that can be applied to both algorithms. A natural way to choose a performance metric is to compare the upfront selected SFC (`USS`) against the target selected SFC (`TSS`). The `TSS` is the SFC that led to the lowest average delay with fixed parameters, after a set of simulations. By ranging one parameter in one chain, we can detect at what values of the varying parameter the `USS` and `TSS` change. Our performance metric is given by the difference of the values at which the `USS` and `TSS` change. In our case the varying parameter is $n^{C,N}$, shown in Pseudocode 1, line 5. Although not time related, we will refer to the values of $n^{C,N}$ at which the `USS` and `TSS` change as the theoretical and target *moment* respectively.

The performance metric is the difference between target and theoretical moments, illustrating how well the algorithms behave under certain circumstances. The performance is better if its value is closer to zero, zero being ideal. A positive value, say k, means that the theoretical moment was '$+k$' moments later than the target moment, in respect to the steps in which the associated parameters varied. A negative value means that the theoretical moment was earlier then the target moment.

5.3 Main Simulation Results

The results of the simulations can be found in Sect. 5.3 for the FB algorithm with SFCs of length one and Sect. 5.3 for the FPB algorithm with SFCs of length two.

The FB Applied to Chains with 1 Node: Expected Duration of Exceeding the Critical Flow Level
Refer to Fig. 5, the results of the simulations are shown whereby parameters are varied as per Pseudocode 1. The variation of the initial values, $n^{i,j}$, (line 1 in Pseudocode 1) is shown on the vertical ax. The variation of the critical levels, $m^{i,j}$, (line 2 in Pseudocode 1) is shown on the horizontal ax. The performance

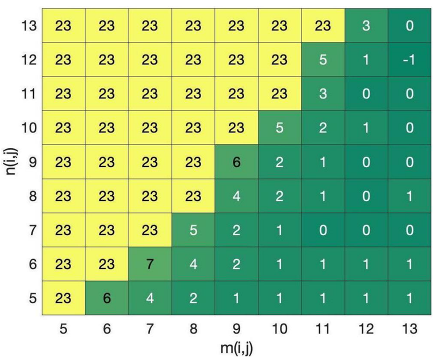

Fig. 5. Performance of FB algorithm, $N = 1$.

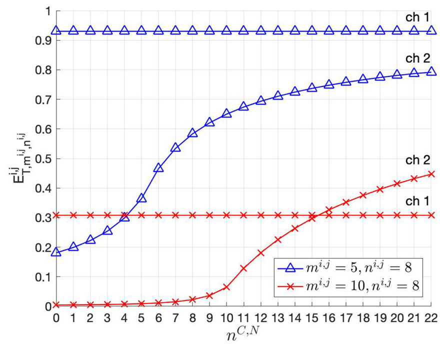

Fig. 6. BG flows exceed critical level, $N = 1$.

of the FB algorithm, resulting from lines 5–7 in Pseudocode 1, is given by the number in the square per $m^{i,j}$, $n^{i,j}$ combination.

Recall that the performance is given by the difference between the theoretical and target moments. The higher this number, the worse the performance. Since the range through which the $n^{C,N}$ varies is limited to $0, 1, \ldots, 22$, '+23' means that no theoretical moment exists within the $n^{C,N}$ range, i.e. the FB algorithm sticks to the same SFC during all the $n^{C,N}$ combinations. The value '+23' could be interpreted as 'off the chart'.

To elaborate more on the decision moments, refer to Fig. 6. The expected fractional time above a critical level, given by $E^{i,j}_{T,m^{i,j},n^{i,j}}$, is shown for two cases, for $n^{i,j} = 8$: $m^{i,j} = 5$ and $m^{i,j} = 10$, each for both chains. For the $m^{i,j} = 10$ case (the '×' graph), the FB algorithm selected chain 2 for $n^{C,N} < 16$ and selected chain 1 for $n^{C,N} \geq 16$, since the expected fractional time above $m^{i,j}$ for chain 2 is lower than the expected fractional time above $m^{i,j}$ for chain 1, for $n^{C,N} < 16$. For the $m^{i,j} = 10$, the theoretical moment lies at 16. From the simulations it follows that the target moment is 14. The performance for this particular case is '+2' and can be found in Fig. 5 for $m^{i,j} = 10$ and $n^{i,j} = 8$.

However, for $m^{i,j} = 5$ (the 'Δ' graph) no change will take place, as the expected fractional time above the critical level for chain 1 is greater than the expected fractional time for chain 2 above the critical level, for all $n^{C,N} = 0, 1, \ldots, 22$. As a result, the FB algorithm will stick to chain 2 and the performance will be '+23', off the chart. This result can be found in Fig. 5 for $m^{i,j} = 5$ and $n^{i,j} = 8$.

The FB algorithm does not perform well in case the initial number BG flows is (roughly) above the critical level as the performance in these circumstances results in a '+23'.

FPB Applied to Chains with Two Nodes

We have applied the FPB algorithm to SFCs consisting of two nodes each, i.e. $N = 2$. The performance results can be found in Fig. 7 whereby the parameters have been varied as per pseudocode 1. The initial values, $n^{i,j}$, are shown on the vertical ax, the critical levels, $m^{i,j}$, are shown on the horizontal ax and the performance of the FPB algorithm, is given by the number in the square per $m^{i,j}$, $n^{i,j}$ combination.

In Fig. 8 the expected duration of BG flows above a critical level (5 and 10) is shown for 13 initial number of BG flows. As per Fig. 8, for critical level 5 and 13 initial BG flows, chain 2 will not be selected for $n^{C,N} = 0, 1, \ldots, 22$. For critical level 10 and 13 initial BG flows, chain 2 is selected for $n^{C,N} \geq 21$.

As opposed to the FB algorithm, the FPB algorithm performs well in overloaded situations for which the initial number of BG flows $n^{i,j}$ are greater than the critical level $m^{i,j}$.

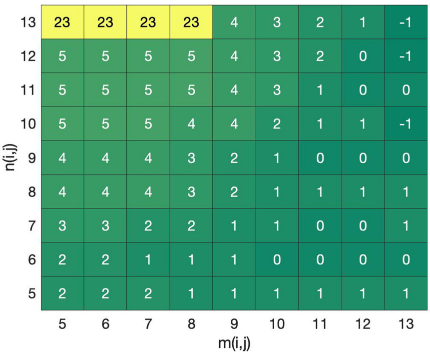

Fig. 7. Performance of FPB algorithm, $N = 2$.

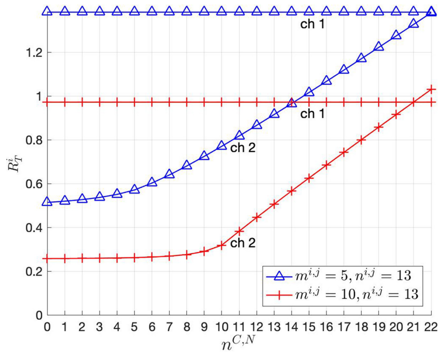

Fig. 8. BG flows exceed critical level, $N = 2$.

6 Conclusion and Future Work

We created two algorithms, a Flow-based and Flow/packet-based algorithm, to select an SFC upon arrival of foreground flow relying only on the flow characteristics and the transient packet behaviour. We consider this a meaning full contribution. In particular the Flow/packet-based algorithm, as it performs well in overloaded situations. The FB algorithm does not perform well in overloaded situations. The number of SFCs and their lengths are limited in real networks and the amount of paths BG traffic can take in a network does not affect the SFC selection.

Adding **machine learning** to the decision algorithms to cope with all the different parameters and decide upfront to assign a flow to an SFC would be the next step in research in dynamic (automated) SFC-RA scheduling.

As mentioned in previous sections, our predictive results are based on exponential distributed parameters, both on a flow and a packet level. In order to work with more **realistic traffic**, results on the expected duration of exceeding a critical level, needs to be extended to other distributions as well. For example, a deterministic arrival rate of packets to simulate voice or video traffic.

References

1. ETSI Industry Specification Group (ISG). Network Functions Virtualisation (NFV); Infrastructure Overview (2015)
2. Haleplidis, E., Pentikousis, K., Denazis, S., Salim, J.H., Meyer, D., Koufopavlou, O.: Software-defined networking (SDN): layers and architecture terminology. In: IETF Network Task Force, no. rfc 7426 (2015)
3. Hantouti, H., Benamar, N., Taleb, T., Laghrissi, A.: Traffic steering for service function changing. IEEE Commun. Surv. Tutor. **21**(1), 487–507 (2018)
4. Halpern, J., Pignataro, C.: Service function chaining (SFC) architecture. In: IETF Network Task Force, no. rfc 7665 (2015)

5. Bhamare, D., Jain, R., Samaka, M., Erbad, A.: A survey on service function chaining. J. Netw. Comput. Appl **75**, 138–155 (2016)
6. Herrera, J.G., Botero, J.F.: Resource allocation in NFV: a comprehensive survey. IEEE Trans. Netw. Serv. Manag **13**(3), 518–532 (2016)
7. Xie, Y., Liu, Z., Wang, S., Wang, Y.: Service function chaining resource allocation: a survey (2016). ArXiv e-prints
8. Kleinrock, L.: Queueing Systems, Volume I: Theory. Wiley & Sons, Hoboken (1975)
9. Ellens, W., Mandjes, M., van den Berg, J., Worm, D., Blaszczuk, S.: Performance valuation using periodic system-state measurements. Perf. Eval. **93**, 27–46 (2015)
10. Wetzels, F., van den Berg, J., Bosman, J., van der Mei, R.: Dynamic priority assignment for SLA compliance in service function chains. In: 26th International Conference on Telecommunications (ICT) (2019)
11. Kaczynski, W., Leemis, L.M., Drew, J.: Transient queueing analyses. INFORMS J. Comput. **24**(1), 10–28 (2012)