# Parallel Multi-path Forwarding Strategy for Named Data Networking

Abdelkader Bouacherine, Mustapha Reda Senouci and Billal Merabti
*A.I. Laboratory, Ecole Militaire Polytechnique,*
*P.O. Box 17, Bordj-El-Bahri 16111, Algiers, Algeria*

Keywords: Named Data Networking, NDN Flow, NDN Fairness, Forwarding Strategy, Forwarding Decisions, Weighted Alpha-proportional Fairness, Flow Assignment Problem.

Abstract: Named Data Networking (NDN) is one of the most promising instantiations of the Information Centric Networking (ICN) philosophy. This new design needs a new thinking due to the fact that the definitions of some concepts used in TCP/IP paradigm are no longer appropriate. In this context, flow and fairness concepts are examined and new perspectives are proposed. An important literature exists about forwarding strategies and congestion control in NDN context. Unfortunately, the lack of definitions pushed many researchers to use the TCP/IP heritage. As a consequence, they neither fully benefit from the native multi-path support nor address the fairness problem. In order to overcome such a drawback and to meet end-users fairness while optimizing network throughput, a new Parallel Multi-Path Forwarding Strategy (PMP-FS) is proposed in this paper. The PMP-FS proactively splits traffic by determining how the multiple routes will be used. It takes into consideration NDN in-network caching and NDN Interest aggregation features to achieve weighted alpha fairness among different flows. Obtained preliminary results show that PMP-FS looks promising.

## 1 INTRODUCTION

Internet is built on redundancy in both communication and information networks. Exploiting this multiplicity is a necessity and an obligation nowadays and in the future Internet (Qadir et al., 2015). With the Information Centric Networking (ICN) proposals, the inherited constraints imposing the single-path routing that limit the utilization of Internet network multiplicity have disappeared. Named Data Networking (NDN) (Jacobson et al., 2009; Zhang et al., 2010; Afanasyev et al., 2014) is a future Internet architecture proposal rolling under the ICN paradigm. The NDN comes with a new communication model based on four main characteristics:

1. Receiver-driven data retrieval model: The user expresses an Interest with a uniquely identified name. The routers use this latter to retrieve the data whose name matches the requested one, and return it to the user;
2. Local state information decisions: They are based on the kept local state information. No global knowledge exists;
3. Loop free forwarding plane: The Interest carries a nonce and the corresponding returned data packet follows the same path in the reverse direction. Thus, none the Interests and data objects can loop for a sufficient period of time;
4. One-to-one flow balance: One Interest brings back at most one data object.

The above communication model led to an adaptive forwarding plane. The latter combined with another innovative feature (i.e. the NDN in-network caching) constitute a platform for multi-path support called ”NDN native multi-path support”. This feature can be used to take advantage of the Internet multiplicity in a parallel (simultaneously) or in a sequence manner (serial) as a backup configuration (detect failure and retry alternative paths). NDN native multi-path support is used to optimize the efficiency (throughput) on one hand, and ensures that end-users get a fair share (fairness) on the other hand.

Serving end-users in a fair manner while maximizing network throughput is fundamental and a challenging task in designing a forwarding plane. The TCP/IP paradigm was built on end-to-end communication model. In this model, the source and the desti-

Abdelkader, B., Senouci, M. and Merabti, B.
Parallel Multi-path Forwarding Strategy for Named Data Networking.
DOI: 10.5220/0005964600360046
In *Proceedings of the 13th International Joint Conference on e-Business and Telecommunications (ICETE 2016) - Volume 1: DCNET*, pages 36-46
ISBN: 978-989-758-196-0

nation are known. Packets in the network are tagged by this information. As a consequence, the fairness between end-users is directly related to the fairness between the flows. With this legacy, we inherited most of the definitions that we use in nowadays communications and networking community.

Identifying the end-users packets in NDN (Interests and data objects) is meant to be impossible in-between within the network. This is the main difference between NDN and the TCP/IP paradigm caused by the disappearance of end-to-end communication model. The NDN architecture is location-independent. Therefore, the NDN Interests can be forwarded individually and independently and data can be duplicated anywhere and so many times in the network (NDN in-network caching feature).

The location-independent nature of NDN, the in-network caching, and the Interest aggregation features brought by NDN are a shifting innovation in networking. This made the rich literature about multi-path forwarding strategies, fairness and congestion control for the TCP/IP architecture no longer appropriate. Although, an important literature exists in NDN context, the lack of definitions forced many researchers to use the TCP/IP heritage. The unpredicted Interest aggregation within the network and the important variation in $RTT$ (Round Trip Time) measurements, as a result of the in-network caching feature, prevent achieving adequate fairness and handling the dynamics in returning data (Yi, 2014). Thus, the native multi-path support has not been taking advantage of.

To design a multi-path forwarding strategy, one has to overcome two barriers. First, it is a necessity to define the NDN flow independently of the source and destination of Interests and data. Second, it is mandatory to maintain end-users fairness and to adopt approaches without relying on $RTT$ measurements.

In this context and in order to benefit of NDN native multi-path support, in this paper, we make the following contributions:

- We explain and clarify ambiguities of the inherited definitions of flow and fairness and we propose new definitions (perspectives) in the context of NDN (section 3).
- We formulate the parallel multi-path forwarding packets as an optimization problem for Coordinated Multi-Path Flow Control (section 4).
- We propose a new localized Parallel Multi-Path Forwarding Strategy (PMP-FS) (section 4). The PMP-FS proactively splits traffic by determining how the multiple routes will be used. It takes into consideration the NDN in-network caching and the NDN Interest aggregation features to achieve weighted alpha fairness among different flows (Mo and Walrand, 2000).
- We report and discuss PMP-FS preliminary results (section 5).

## 2 BACKGROUND AND RELATED WORK

Internet is built on redundancy. It is a system built on shared resources. As a system, it seeks to optimize its efficiency (throughput) on one hand, and because of its sharing nature, it seeks to ensure that end-users get a fair share (fairness), on the other hand.

The native support of parallel multi-path forwarding in NDN is to be taken advantage of. Indeed, the NDN Forwarding Information Base (FIB) entry contains a list of ranked next-hops for a specific name prefix to a fetched data object with a unique name. This data object can be duplicated (cached) anywhere and so many times in the network. These characteristics lead to a competition between outgoing Interests as to be forwarded to the set of best possible face(s) while preserving fairness between end-users.

### 2.1 Background

Fairness is a crucial concept in networking that needs complementary information to be understood, since it varies from equality to equity and can be defined in many ways. It has therefore already been the subject of intensive research. Many definitions were given such as Weighted Proportional Fairness (Kelly, 1997), Proportional Fairness (Mazumdar et al., 1991), and the Max-Min Fairness (Hahne, 1991). A unifying mathematical model to fair throughput was first introduced in (Mo and Walrand, 2000). The so-called alpha-fair utility functions $U_s(x_s)$ (Equation 1), which defines the different notions of fairness depending on the choice of a parameter $\alpha$. This latter is a kind of degree of fairness that controls the trade-off between efficiency and fairness. Please refer to Table 1 for notations.

$$U_s(x_s) = \begin{cases} w_s^{\alpha} \ \frac{x_s^{1-\alpha}}{1-\alpha} \ , if \ \alpha \neq 1 \\ \\ w_s \log(x_s) \ , if \ \alpha = 1 \end{cases} \tag{1}$$
$$for \, w > 0, \ \alpha \geq \ 0, \ l \in L$$

For example, $\alpha = 0$ with $w = 1$ corresponds to the system maximum efficiency or throughput, $\alpha \ \rightarrow \ \infty$ with $w = 1$ corresponds to the Max-Min fairness, and $\alpha = 2$ with $w = \frac{1}{RTT^2}$ corresponds to the TCP fair (Low, 2003).

Table 1: Notations.

| Notation | Meaning |
|---|---|
| $L$ | set of resources (arcs) |
| $P$ | set of paths or routes |
| $l \in P$ | resource $l$ is on path $p$ |
| $C_l$ | capacity of resource $l$ |
| $AX \leq C$ | capacity constraints |
| $S$ | set of flows |
| $s$ | individual flow (session) |
| $w_s$ | weight of flow (session) s |
| $x_s$ | rate of flow (session) s |
| $y_p$ | flow rate on path $p$ |
| $U_s(x_s)$ | alpha-fair utility functions |
| $q$ | a number close but less than 1 |
| $\alpha$ | a parameter reflecting the fairness |
| $fl$ | number of active flows |
| $\mu_l$ | shadow price, or rate of congestion indication at resource $l$ |

The proposed fairness schemes used in the TCP/IP paradigm with enough bandwidth demand allocates equal bandwidth to the active flows. As a consequence, the fairness between end-users is directly related to the fairness between the flows. It has worked because we had a sharp definition of the flow. It is not the case of NDN.

## 2.2 Related Work

NDN differs mainly from the IP paradigm by the disappearance of the end-to-end communication model. NDN is source-destination free, where NDN routers have no idea of the provenance of Interests and their possible destinations. Speaking of end-users fairness passes certainly through a clear definition of the NDN flow.

In (Wang, 2013), the author gives a definition of NDN flow as triplet [( group of Interests, data fetched by this group), client, name-prefix]. He supposes that the name-prefix is indivisible in the routing/forwarding domain and the Interests are not forwarded individually. The author claims that the difficulty of defining a flow is not due to the multi-homing and NDN in-caching feature. We do not agree with the client component, since NDN routers ignore completely the sources and destinations of the traffic. We claim as the authors of (Yi et al., 2012; Yi et al., 2013; Yi, 2014) did, that fairness in NDN context can be only defined by the names prefixes carried by the packets.

Let us take the definition of a flow presented in (Wang, 2013) and the max-min fairness presented in (Yi, 2014). At the equilibrium state, the max-min fairness allocates equal bandwidth to the active flows. Let us have the example of ”www/com/Facebook/” as a name prefix or as a delegation name (Afanasyev et al., 2015a), representing one FIB entry under which we have 9999 pending Interests. The latter name prefix will get the same bandwidth share with another name prefix of 99 pending Interests. Most researchers might consider, as we do, this equality as unfair. Defining the NDN flow to maintain end-users fairness is mandatory. In the next section, we present definitions of flow and fairness in the context of NDN.

# 3 DEFINITIONS

To define an NDN flow two questions should be answered:

1. How NDN routers treat names?
2. How NDN routers treat different types of data?

To answer the first question, let us examine the NDN Forwarding Information Base (NDN FIB) functionalities. The NDN FIB entry contains a list of ranked next-hop for specific name prefix. Due to scalability issues (Baid et al., 2012; Narayanan and Oran, 2015), consensus has not yet been reached on how and what prefix names to populate the FIB. They are a subject of intensive research (Afanasyev et al., 2015b; Afanasyev and Wang, 2015; Song et al., 2015; Yuan et al., 2012). All mentioned proposals advocate that NDN packet is forwarded by performing a look-up of its content name using the Longest Prefix Matching (LPM) or the Longest Prefix Classification (LPC). The different Interests with the same routable prefix name issued by different end-users are forwarded under the same FIB name-prefix. Furthermore, the Interests forwarded under the same FIB entry at one router might be forwarded under different FIB entries at another router, and therefore could be forwarded separately.

To answer the second question, we examined the NDN Pending Interest Table (NDN PIT) functionalities. As a core component, PIT is responsible for keeping track of the awaiting Interest packets (Yi et al., 2012; Yuan and Crowley, 2014). An Interest is issued by one end-user but it could be aggregated (becoming subsequent Interest) once or many times at the downstream routers. Therefore, for a given router within the network, a pending Interest could be issued by one or many end-users. A returned data object could satisfy one or multiple end-users. The Interest aggregation feature combined with other features (NDN in-network caching, adaptive forwarding plane, ...) make the power of the NDN architecture. It is important to notice that Interests to be aggregated should have the exact same prefix name and generally this concerns the static content or a Content Delivery Networks (CDN) content. Another aspect to consider is the fact that under the same routable name (prefix-name), we can find a big number of different distinct Interests issued by one or more than one end-user.

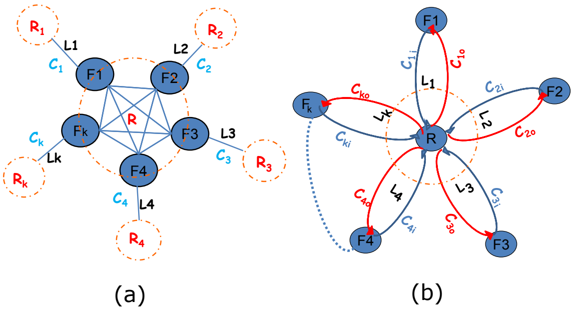
From the above discussion, it is clear that defining a flow should be local to the router and for a specific period of time (FIB update). In order to have a clear idea, we model a Router $R$ and its links with its neighbor routers by a weighted directed graph $G$.

## 3.1 Router Model

A router $R$ is connected to its $K^{ith}$ neighbor router by a link $L_k$ with capacity $C_k$, through the corresponding face $F_k$. The faces of the router $R$ are completely connected to each other with infinite capacity links (Figure 1-a).

We model the router and its direct links as a directed graph $G = (F, L, C)$ (Figure 1-b) where:

- $F$ is a set of vertices (Router faces and a virtual central node $R$);
- $L$ is the set of edges (Router links);
- $C$ is a function whose domain is $L$, it is an assignment of capacities to the edges (Router links in terms of inputs and outputs bandwidth);
- The order of graph is $k+1$ and the size of $G$ is $|L(G)| = 2k$;
- $L_{ki}$ denotes the arc going from $F_k$ to R, represents the downstream through the $K^{ith}$ face $F_k$, with a capacity $C_{ki}$;
- $L_{ko}$ denote the arc going from $R$ to $F_k$ , represents the upstream through the $K^{ith}$ face $F_k$, with a capacity $C_{ko}$;
- $C_{ki} + C_{ko} <= C_k$, is the capacity of bidirectional link $k$ connecting the router $R$ and the $K^{ith}$ neighbor router, through the face $F_k$ (Figure 1).



(a) (b)

Figure 1: Router Model.

**Definition 3.1.** A Flow in NDN context is a set of Interests and their corresponding returned data objects forwarded under the same FIB entry proper to a router during a given period. These Interests may have been requested from one or multiple faces and forwarded through one or multiple faces (Figure 2).

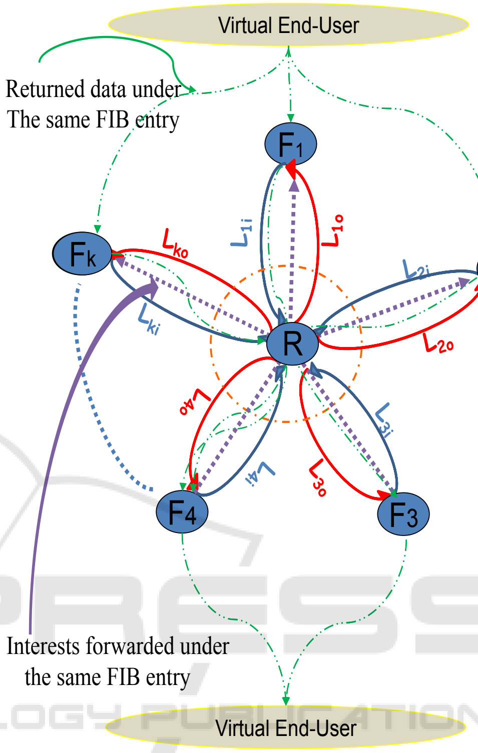


Figure 2: NDN Flow.

**Example of NDN Flow:** Let us consider an example from Figure 2. In case the Interests were received from face $F_3$ and $F_4$, the FIB lookup mechanism choses for them the same FIB entry with the corresponding FIB outgoing faces $F_1$,$F_2$ and $F_k$. We consider, with analogy to the connected oriented model:

- A virtual end-user source makes connections (max. of six, three in our example) to the other virtual end-user destination;
- The tuple (virtual end-user source, $F_3$, $R$, $F_2$, virtual end-user destination) is a connection;
- If an Interest is forwarded through face $F_2$ and it is received from face $F_3$, the Interest will use the resources $l_{3i}$ through $R$ and $l_{2o}$;
- The returned data will follow the reverse path $l_{2i}$ through $R$ and $l_{3o}$.

This set of Interests and their corresponding returned data objects, are considered for this router as one NDN flow.

**Definition 3.2.** In the presence of competing elastic flows, a Flow Assignment is fair in NDN context if the assignment is weighted alpha-fair, where:

- Weights represent the normalized number of distinct awaiting sub-interests and the answered Interests from the Content Store (CS) as a function of time.
- Distinct data object packets sizes and heterogeneous round-trip times ($RTT$s) are taken into consideration.

## 3.2 Discussion

The fairness must be addressed across flows, across network and across time (Briscoe, 2007). The proposed definition of NDN flow and the fairness criteria does not specify whether many flows can serve a common end-user. We claim in this paper a realistic user-centric conception of fairness. The weighted $\alpha$-proportional fairness imposed between virtual end-users flows at each node ensures fairness between the real end-users. The notion of friendliness imposed by the emergence of the peer-to-peer networks in the context of TCP/IP paradigm does not need to be addressed in the context of NDN. Taking the definitions above, all the applications and sessions are friendly to each other if the flows are fairly tackled.

In (Yi, 2014), the authors present Fair Interest Limiting (FIL). An NDN version of the fair queuing mechanism used in the TCP/IP paradigm that suffers from the same limitations in terms of fairness (Briscoe, 2007). Besides, it does neither take the unpredictable effect of content in-network caching and the Interest aggregation feature nor the distinct data object packets sizes and heterogeneous round-trip times ($RTT$s) into consideration on the fairness evaluation. At the opposite, weights presented in the above definition come as an answer. The weights must be a function of time to deal with the dynamic nature and unpredictable changes in the network caused by the effect of in-network caching and the Interest aggregation features.

# 4 PARALLEL MULTI-PATH FORWARDING STRATEGY

Internet has a multi-path nature. This latter was not exploited by the inherited Internet technologies although its benefits in terms of resilience and resource usage efficiency (Qadir et al., 2015). The efforts of deploying an Internet-scale multi-path protocol were facing the end-to-end communication model. Exchanging path information and the large forwarding tables lead to computational and storage overhead in large-scale networks. In this context, scalability was the black hole (Qadir et al., 2015; Baid et al., 2012).

In this section, we diagnose the problem, outline the research problems and design considerations. Further, we fix the objectives and we propose a new forwarding strategy.

## 4.1 Diagnosis: The Nature of the Challenge

The future Internet is definitely multi-path (Qadir et al., 2015). In this context, we propose the PMP-FS which is a multi-path strategy that takes advantage of NDN native support of parallel multi-path forwarding. It aims to serve end-users in a fair manner without the need of identifying them while maximizing network throughput. Here, it is significant to mention:

- The requested data objects in NDN may be cached in different locations and in different periods of time by the multi-homing mechanism and/or the NDN in-caching mechanism. The whole data objects may be in one place or dispersed in different places;
- The order of data packets arrival constitutes another difference in forwarding packets between NDN and TCP/IP paradigm. The order of data packets arrival is very important in TCP/IP, which is not the case in NDN. The returned data will be recomposed for application use by end-users;
- Each flow (session) aims at maximizing its throughput;
- Resources are limited: the competition between flows as to be better served by forwarding them to the set of the best possible face(s) is unavoidable;
- Preserving not only fairness between end-users but also efficiency and scalability are a requirement;
- PMP-FS is to be executed at every NDN router;
- PMP-FS should respect the real time (line-speed) constraint;
- PMP-FS should take the effect of content in-network caching mechanism and the subsequent awaiting Interests on the evaluation of fairness;
- Hop-by-hop Interest shaping mechanism to ensure the whole network stability is necessary;
- We assume that the flows are elastic, the routing plane is responsible for populating the FIB and we

make no assumptions about whether the paths are disjoint.

## 4.2 Guiding Policy: Dealing with the Challenge

In (Yi et al., 2012; Yi et al., 2013; Yi, 2014), the authors use multi-path feature that NDN offers for recovering from failures (detect failure and retry alternative path) and divert excess traffic to other paths which we call serial multi-path. This latter approach ignores the competing nature of flows. Oppositely, parallel multi-path uses all or part of the available paths in a controlled way to forward the Interests of a flow simultaneously over this set of available paths.

The PMP-FS splits traffic and achieves fairness in NDN context of the active flows over multiple different faces at each time slot. This increases reliability, robustness and fault tolerance. The PMP-FS works as follows:

- A hop-by-hop Interest shaping module proposed in (Wang et al., 2013) is used as a congestion control mechanism. It is to be executed at every NDN router to ensure the whole network stability. The result of this module is the links input and output capacities between the router at hand and its neighbor routers;
- At every new FIB entry picked:
  1. A new flow queue created and set as inactive;
  2. The first group of Interests ($f$ Interests) is forwarded immediately;
  3. The flow queue becomes active when the forwarded Interests bring back data. The size of data object is estimated;
  4. Proportional Integral Controller Enhanced (PIE) is used (Pan et al., 2013).
- At every data object packet received or cache hit:
  1. Increment the per flow counter.
- At every time slot:
  1. the PMP-FS collects:
     - The vector of per flow counters;
     - The information about the pending Interests and the awaiting Interests for each active NDN flow and their corresponding FIB entries;
     - The links input and output capacities: the result of the hop-by-hop Interest shaping capacities minus the reserved bandwidth for the ending sessions (empty flows queues).
  2. Flow Assignment module is executed for the active flows queues: the result is a matrix $A$ of Flow Assignment;
  3. Forwarding Decision module is executed:
     - Taking as input:
       - the matrix $A$ of Flow Assignment of the last step;
       - the vector of the number of the pending Interests;
       - the vector of estimated sizes of data objects of the active flows;
     - The output is a matrix $D$ of the number of Interests to forward for each flow over each face;
- The matrix $D$ is used to forward the incoming Interests and for every Interest answered another one from the same flow is sent.

  The time slot must be chosen in a manner that enable us to integrate the new flows softly and eliminate the $RTT$ variations effect on fairness. We take 25 *msec* as the time slot which is about the third of mean Internet $RTT$.

## 4.3 Action Plan: Carrying out the Guiding Policy

NDN follows the pull-based model. Interests come through one of the faces. Based on the FIB, these Interests are forwarded through other face(s). The returned data follows the reverse direction. The global architecture of PMP-FS is depicted in Figure 3.

- At every new FIB entry picked:
  - When a router receives an Interest and in case the checking of the CS and the PIT results in a negative response, the FIB is checked. At this moment a new FIB entry is picked and the $after\ receive\ Interest$ action is triggered.
  - The variation in data object sizes is taken into consideration by the per-flow queuing (counters per name-space) and the flow will be considered by the Flow Assignment module only when it is active. In this case, we have the estimated size of the data object packets. The latter is smoothly updated at each received data object (the $before\ satisfy\ Interest$ action is triggered).
  - Forwarding the first group of Interests of a newly created flow over a set $f$ of faces gives us an opportunity to measure the performance and rank the faces both dynamically and locally ($f$ best faces ranked by the routing plane).
  - Proportional Integral Controller Enhanced (PIE) is used as an acceptance mechanism by dropping incoming Interests and sending pack NACKs based on a probability. Departure rate and the flow queue length are used

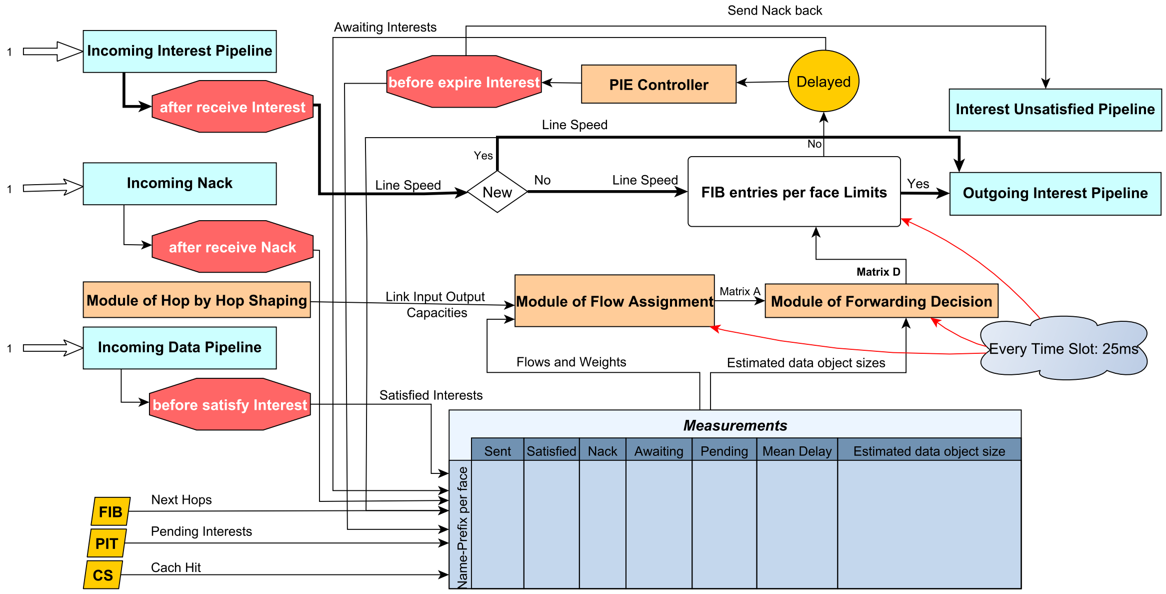


Figure 3: PMP-FS Global Architecture.

(Pan et al., 2013). When sending NACKs the *before expire Interest* action is triggered to update the counters in measurements table.

- At every time slot (every 25 *msec*)

1. the PMP-FS collects:
   - The information about the pending and the awaiting Interests with their corresponding FIB entries for each active NDN flow and the Interests that hit the cache. These will be used as weights;
   - The links input and output capacities: the result of the capacities of the hop-by-hop shaping minus the reserved space for the ending sessions (empty flows queues);
2. The Flow Assignment module is executed for the active flows. The result is a matrix $A$ of Flow Assignment over the faces. The objective of this module is to perform a controlled splitting of the active flows over the faces while satisfying the limited bandwidth constraints of the outgoing and ingoing router links. In the next section, the problem mathematical formulation is presented.
3. The module of Forwarding Decision: The one-to-one interdependence between Interests and data objects called NDN flow balance propriety gives us the opportunity to carry out a controlled splitting of the flows over the faces by only deciding where and how many Interests of the active flows to forward.

   Taking as inputs the matrix of Flow Assignment of the last step and the vector of estimated sizes of the active flows data objects, we get the matrix $D$ of the maximum number of Interests to be forwarded for each flow over each face.

   The matrix $D$ is used to forward the incoming Interests, the next time slot and for every Interest answered another one from the same flow is sent.

### 4.3.1 Network Model and Problem Formulation

To design The Flow Assignment module of the PMP-FS, we use the model described previously (cf. subsection 3.1). We use the directed graph $G = (F,L,C)$ (cf. Figure 1).

For a better understanding, let us take the example of the NDN flow $fw$ (see Figure 2): if Interests of the same flow enter through face $F_3$ and $F_4$, their corresponding FIB outgoing faces are face $F_1$, $F_2$ and $F_k$. If an Interest of flow $fw$ is forwarded through face $F_2$ and it is received from face $F_3$. The latter Interest would use the resources $l_{3i}$ through $R$ and $l_{2o}$. The returned data object will follow the reverse path, $l_{2i}$ through $R$ and $l_{3o}$.

This flow $fw$ is competing with other flows to grab the max of bandwidth. The links capacities are the result of the hop-by-hop Interest shaping mechanism. The issue will be an optimization problem that takes into consideration only the paths and seizes of returned data objects.

The Flow Assignment module will decide the splitting of active flows (sessions) over the paths while satisfying the limited bandwidth constraints. In other words, the module gives a solution of an optimization problem for a weighted alpha fair flow assignment. It takes into consideration the paths and seizes of returned data objects only. The result of this optimization will be used to decide on the number of Interests of each flow to be forwarded on each face.

Real time was always an issue when solving the Flow Assignment and the Coordinated Multi-Path Flow Control type of problems. The computation scalability led to the use of off-line methods. The authors in (McCormick et al., 2014) propose an algorithm that could be executed in the range of milliseconds and could be used as an on-line tool. It is proposed as a Software Defined Networking (SDN) Traffic Engineering tool based on the work presented in (Voice, 2007). Oppositely to (McCormick et al., 2014), we use the proposed algorithm in a distributed manner to achieve weighted alpha-fairness flow assignment at each router of the network. We apply the algorithm in our proposed network model (cf. Figure 1).

The Flow Assignment optimization using weighted alpha-fair utility functions is defined as follows:

$$Maximize \quad \sum_{s\in S} w_s^{\alpha} \; \frac{X_s^{1-\alpha}}{1-\alpha} \tag{2}$$

$$Subject\ to \begin{cases} \sum\limits_{p\in l} y_p \leq C_l \\ \sum\limits_{p\in s} y_p = x_s \end{cases} \tag{3}$$

$$for\, w > 0,\ \alpha \neq 1, x > 0,\ y > 0,\ l \in L$$

The resulting updating rules, following (McCormick et al., 2014; Kelly et al., 2008; Voice, 2007), using the same notations as in Table 1 are:

$$y_p = \left(\left(\tfrac{w_{s(p)}}{x_{s(p)}}\right)^{\alpha} \tfrac{1}{\sum\limits_{l\in p}\mu_l}\right)^{\frac{1}{1-q}} x_{s(p)} \tag{4}$$

$$\mu_l(t+1) = \mu_l(t) + \tfrac{1-q}{2}\mu_l(t) \left[ \frac{\sum\limits_{p\in l} y_p(t) - c_l}{c_l} \right] \tag{5}$$

$$x_s(t+1) = x_s(t) + \tfrac{1-q}{2(\alpha+q-1)} x_s(t) \left[ \frac{\sum\limits_{p\in s} y_p(t)^q - x_s(t)^q}{x_s(t)^q} \right] \tag{6}$$

The main advantage of this formulation is that the updating rules can be implemented in parallel. The $\mu_l(t)$ is the shadow price of the *link l* at iteration $t$ of the *path p*.

In the sequel, we present some preliminary results. Furthermore, we discuss implementation and evaluation considerations.

# 5 PERFORMANCE EVALUATION

The general structure of the module describing how we can implement the formulation described in subsubsection 4.3.1 is shown in algorithm 1.

Algorithm 1: Flow Assignment Module.

```
Input: G = (F,L,C), Weights $W[fl]$.
begin for All Flows s do
    for All Sub Flows of s do
        for All Sub Flows $Y_p$ do
            $y_p \leftarrow ((\frac{w_{s(p)}}{x_{s(p)}})^{\alpha} \frac{1}{\sum_{l\in p}\mu_l})^{\frac{1}{1-q}} x_{s(p)}$
        repeat
            for All links l do
                $\mu_l \leftarrow \mu_l + \frac{1-q}{2}\mu_l \left[\frac{\sum_{p\in l} y_p - c_l}{c_l}\right]$
            for All Aggregate Flows $X_s$ do
                $x_s \leftarrow x_s + \frac{1-q}{2(\alpha+q-1)} x_s \left[\frac{\sum_{p\in s} y_p^q - x_s^q}{x_s^q}\right]$
            $Ypold \leftarrow y_p$
            for All Sub Flows $Y_p$ do
                $y_p \leftarrow ((\frac{w_{s(p)}}{x_{s(p)}})^{\alpha} \frac{1}{\sum_{l\in p}\mu_l})^{\frac{1}{1-q}} x_{s(p)}$
        until $(|y_p - Ypold| < 10^{-5})$
return Matrix of Flow Assignment A
```

## 5.1 Preliminary Results

An implementation of algorithm 1 in C++ on an Intel Core i5-3550 (6M Cache, 3.3 GHz) processor gives the results shown in Figure 4.

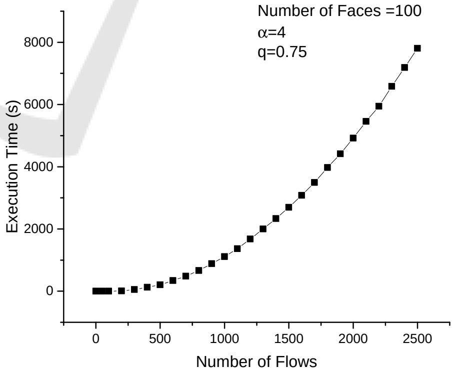


Figure 4: Execution times per number of flows without parallelism.

The execution time grows exponentially with the number of flows, which explains the real-time issue that was previously mentioned. In Figure 5, we studied the effect of number of faces on the execution time. Obtained results show that the number of faces does not affect the execution time.

The Field-programmable Gate Array (FPGA) implementation of algorithm 1 on *BT*21*CN* network

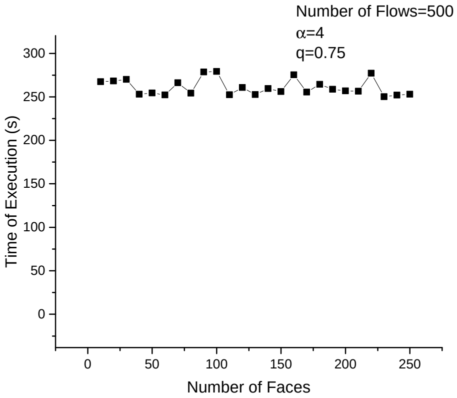


Figure 5: Execution times per number of faces without parallelism.

comprising 106 nodes and 234 links was done by (McCormick et al., 2014) on a Virtex 7 FPGA. The simulation was run with more than ten thousand (10000) flows in less than two (2) *msec* (McCormick et al., 2014). If we have a router with hundred (100) faces, following our model we would have one hundred and one (101) nodes and two hundred (200) links network. The $RTT$ in todays Internet is around 80 *ms* on average (Perino and Varvello, 2011). We take 25 *msec* as the time slot, which is about the third of Internet $RTT$ average. With the latter frequency of execution of the Flow Assignment module (three times in one $RTT$ round), PMP-FS can integrate the new flows softly and eliminate the $RTT$ variations effect on fairness. This proves the computation time scalability of our PMP-FS.

## 5.2 Discussion

The actual configuration of the FPGA implementation of (McCormick et al., 2014) supports 512 links, 32000 flows and 96K paths. Using our model, the latter configuration is equivalent to a router with 256 faces and 32000 NDN flows.

The number of active flows in a core router is in the order of $10^3$ (Mills et al., 2010), which makes the PMP-FS an excellent choice for use on Internet scale network. But it is significant to mention that one has to design for the future. We argue, besides the technical advances in the future that would make the module limits higher, our proposal is configurable. It is in the same spirit as (Afanasyev and Wang, 2015). The use of hierarchical names by NDN made our proposal flexible to the level of granularity. Maintaining per-flow states information is undesirable in large networks but in the case of PMP-FS one has to see it as a class-level or flow aggregation with different granularities. Setting the degree of granularity (the level of the FIB entry in the name tree to consider by the Flow Assignment Module) depend on the role of the router in the network (Edge or Core Router). The latter flexibility makes the PMP-FS adaptable to support any exponential growth of Internet. In this context, it is worth highlighting that:

- The choice of the degree of granularity combined with the weights of NDN flows makes PMP-FS able to handle any number of flows. The fairness would not be affected and can be controlled by the choice of the degree of granularity;
- The degree of granularity is used to reduce the per-flow state managed by each router and to control the processing time and storing space;
- The weights of NDN flows can be used to support differentiated bandwidth allocation (based on priority);
- The in and out link capacities used by the Assignment Flow Module can be used and configured to leave room in the links for QoS guaranteed applications.
- The packets (Interests and Data Objects) are forwarded in line speed.

# 6 CONCLUSION AND FUTURE WORK

In this paper, we have proposed a new Parallel Multi-Path Forwarding Strategy (PMP-FS) that takes into consideration NDN in-network caching and NDN Interest aggregation features to achieve weighted alpha fairness among different NDN flows. In one hand, the proposed PMP-FS is a general framework for handling fairness and throughput (by setting the tuning parameters (e.g. $\alpha$ and $q$)). In the other hand, it can handle all type of application needs (delay, bandwidth, ...) and communication protocols (over TCP, Ethernet, ...). Besides it can be used to implement differentiated services by routable name-prefix providing quality of service (QoS). We do believe that the major drawback of the proposed solution is the need of an extra hardware. However, we do believe also that it will add value of robustness and efficiency in the future Internet architecture.

The evaluation of the PMP-FS in the ndnSIM 2.1, an open-source simulator for NDN (Afanasyev et al., 2012; Mastorakis et al., 2015), is a work-in-progress. In this context, it is worth pointing out that the forwarding strategies implemented in the ndnSIM are special cases of the proposed PMP-FS strategy:

- Best Route Strategy: PMP-FS with one best link;

- Multi-cast Strategy: PMP-FS with all upstreams indicated by FIB entry;
- Client Control Strategy: PMP-FS testing specific Interests and allowing them to be forwarded to preconfigured outgoing faces.

## REFERENCES

Afanasyev, A., Burke, J., Zhang, L., Claffy, K., Wang, L., Jacobson, V., Crowley, P., Papadopoulos, C., and Zhang, B. (2014). Named Data Networking. *ACM SIGCOMM Computer Communication Review*, 44(3):66–73.

Afanasyev, A., Moiseenko, I., and Zhang, L. (2012). ndnSIM: NDN simulator for NS-3. Technical report, NDN-0005.

Afanasyev, A., Shi, J., Zhang, B., Zhang, L., Moiseenko, I., Yu, Y., Shang, W., Huang, Y., Abraham, J. P., Dibenedetto, S., Fan, C., Pesavento, D., Grassi, G., Pau, G., Zhang, H., Song, T., Abraham, H. B., Crowley, P., Amin, S. O., Lehman, V., and Wang, L. (2015a). NFD Developer ' s Guide. Technical report, NDN-0021.

Afanasyev, A. and Wang, L. (2015). Map-and-Encap for Scaling NDN Routing. Technical report, NDN-0004.

Afanasyev, A., Yi, C., Wang, L., Zhang, B., and Zhang, L. (2015b). SNAMP: Secure Namespace Mapping to Scale NDN Forwarding. *IEEE Global Internet Symposium (GI 2015)*.

Baid, A., Vu, T., and Raychaudhuri, D. (2012). Comparing alternative approaches for networking of named objects in the future Internet. In *Computer Communications Workshops (INFOCOM WKSHPS), 2012 IEEE Conference on Emerging Design Choices in Name-Oriented Networking*, pages 298–303.

Briscoe, B. (2007). Flow rate fairness: Dismantling a religion. *ACM SIGCOMM Computer Communication Review*, 37(2):63–74.

Hahne, E. L. (1991). Round-robin scheduling for max-min fairness in data networks. *Selected Areas in Communications, IEEE Journal on*, 9(7):1024–1039.

Jacobson, V., Smetters, D. K., Briggs, N. H., Plass, M. F., Stewart, P., Thornton, J., and Braynard, R. L. (2009). VoCCN: Voice-over Content-Centric Networks. *Proceedings of the 2009 workshop on Re-architecting the internet*, pages 1–6.

Kelly, F. (1997). Charging and rate control for elastic traffic. *European Transactions on Telecommunications*, 8(1):33–37.

Kelly, F., Raina, G., and Voice, T. (2008). Stability and fairness of explicit congestion control with small buffers. *ACM SIGCOMM Computer Communication Review*, 38(3):51.

Low, S. H. (2003). A duality model of TCP and queue management algorithms. *IEEE/ACM Transactions on networking*, 11(4):525–536.

Mastorakis, S., Afanasyev, A., Moiseenko, I., and Zhang, L. (2015). ndnSIM 2 . 0 : A new version of the NDN simulator for NS-3. Technical report, NDN-0028.

Mazumdar, R., Mason, L. G., and Douligeris, C. (1991). Fairness in network optimal flow control: optimality of product forms. *Communications, IEEE Transactions on*, 39(5):775–782.

McCormick, B., Kelly, F., Plante, P., Gunning, P., and Ashwood-Smith, P. (2014). Real time alpha-fairness based traffic engineering. *Proceedings of the third workshop on Hot topics in software defined networking - HotSDN '14*, 1(1):199–200.

Mills, K., Filliben, J., Cho, D., Schwartz, E., and Genin, D. (2010). Study of Proposed Internet Congestion Control Algorithms. Technical report, NIST Special Publication 500-282.

Mo, J. and Walrand, J. (2000). Fair end-to-end Window-based Congestion Control. *IEEE/ACM Trans. Netw.*, 8(5):556–567.

Narayanan, A. and Oran, D. (2015). Ndn and Ip Routing Can It Scale ? In *Proposed Information-Centric Networking Research Group (ICNRG), Side meeting at IETF-82, Taipei*.

Pan, R., Natarajan, P., Piglione, C., Prabhu, M. S., Subramanian, V., Baker, F., and VerSteeg, B. (2013). PIE: A lightweight control scheme to address the bufferbloat problem. In *14th International Conference on High Performance Switching and Routing (HPSR), IEEE*, pages 148–155, Taipei, Taiwan.

Perino, D. and Varvello, M. (2011). A reality check for content centric networking. In *Proceedings of the ACM SIGCOMM workshop on Information-centric networking - ICN '11*, page 44, New York, USA. ACM Press.

Qadir, J., Ali, A., Yau, K.-l. A., Sathiaseelan, A., and Crowcroft, J. (2015). Exploiting the power of multiplicity: a holistic survey of network-layer multipath. *arXiv:1502.02111v1 [cs.NI]*, pages 1–35.

Song, T., Yuan, H., Crowley, P., and Zhang, B. (2015). Scalable Name-Based Packet Forwarding: From Millions to Billions. In *Proceedings of the Second International Conference on Information-Centric Networking*, ICN '15, pages 19–28, New York, USA. ACM.

Voice, T. (2007). Stability of multi-path dual congestion control algorithms. *IEEE/ACM Transactions on Networking*, 15(6):1231–1239.

Wang, Y. (2013). *Caching, Routing and Congestion Control in a Future Information-Centric Internet*. PhD thesis, North Carolina State University.

Wang, Y., Rozhnova, N., Narayanan, A., Oran, D., and Rhee, I. (2013). An improved hop-by-hop interest shaper for congestion control in named data networking. *ACM SIGCOMM Computer Communication Review*, 43(4):55–60.

Yi, C. (2014). *Adaptive forwarding in named data networking*. PhD thesis, The University Of Arizona.

Yi, C., Afanasyev, A., Moiseenko, I., Wang, L., Zhang, B., and Zhang, L. (2013). A Case for Stateful Forwarding Plane. *Computer Communications*, 36(7):779–791.

Yi, C., Afanasyev, A., Wang, L., Zhang, B., and Zhang, L. (2012). Adaptive forwarding in named data networking. *ACM SIGCOMM Computer Communication Review*, 42(3):62.

Yuan, H. and Crowley, P. (2014). Scalable Pending Interest Table design: From principles to practice. *Proceedings - IEEE INFOCOM*, pages 2049–2057.

Yuan, H., Song, T., and Crowley, P. (2012). Scalable NDN Forwarding: Concepts, Issues and Principles. *21st International Conference on Computer Communications and Networks (ICCCN)*, pages 1–9.

Zhang, L., Estrin, D., Burke, J., Jacobson, V., Thorton, J. D., Smetters, D. K., Zhang, B., Tsudik, G., Claffy, K., Krioukov, D., Massey, D., Papadopoulos, C., Abdelzaher, T., Wang, L., Crowley, P., and Yeh, E. (2010). Named Data Networking. Technical report, NDN-0001.