UNIVERSIDADE FEDERAL FLUMINENSE

DALBERT MATOS MASCARENHAS

LIMITING INTEREST-PACKET FORWARDING IN CONTENT-CENTRIC WIRELESS MESH NETWORKS

NITERÓI

2017

UNIVERSIDADE FEDERAL FLUMINENSE

DALBERT MATOS MASCARENHAS

LIMITING INTEREST-PACKET FORWARDING IN CONTENT-CENTRIC WIRELESS MESH NETWORKS

Tese de Doutorado apresentada ao Programa de Pós-Graduação em Computação da Universidade Federal Fluminense como requisito parcial para a obtenção do Grau de Doutor em Computação. Área de concentração: Sistemas de Computação

Orientador:
IGOR MONTEIRO MORAES

NITERÓI

2017

Ficha Catalográfica elaborada pela Biblioteca da Escola de Engenharia e Instituto de Computação da UFF

M395 Mascarenhas, Dalbert Matos
Limiting interest-packet forwarding in content-centric wireless mesh networks / Dalbert Matos Mascarenhas. – Niterói, RJ : [s.n.], 2017.
91 f.

Tese (Doutorado em Computação) - Universidade Federal Fluminense, 2017.
Orientador: Igor Monteiro Moraes.

1. Arquitetura de rede (Computação). 2. Roteamento. 3. Rede sem fio. 4. Rede Mesh. I. Título.

CDD 004.65

DALBERT MATOS MASCARENHAS

LIMITING INTEREST-PACKET FORWARDING IN CONTENT-CENTRIC WIRELESS MESH NETWORKS

Tese de Doutorado apresentada ao Programa de Pós-Graduação em Computação da Universidade Federal Fluminense como requisito parcial para a obtenção do Grau de Doutor em Computação. Área de concentração: Sistemas de Computação

Aprovada em agosto de 2017.

BANCA EXAMINADORA

Igor Monteiro Moraes

Prof. Igor Monteiro Moraes, D.Sc. - Orientador, UFF

Prof. Débora Christina Muchaluat Saade, D.Sc., UFF

marcelo gonçalves Rubinstein

Prof. Marcelo Gonçalves Rubinstein, D.Sc., UERJ

Prof. Artur Ziviani, Dr., LNCC

Prof. Célio Vinicius Neves de Albuquerque, Ph.D., UFF

Niterói

2017

Agradecimentos

Agradeço à minha família, minhã mãe Adalgisa, meu pai Gilberto e minha esposa Juliana por todo apoio e paciência durante esse tempo.

Agradeço também aos professores da UFF, em especial ao meu orientador Igor Monteiro Moraes por todo o apoio, dedicação, conselhos e contribuição. Agradeço também à professora Débora Christina Muchaluat Saade por todo apoio e incentivo. Agradeço ao professor Marcelo Gonçalves Rubinstein por ter aceitado participar da banca examinadora e por ter contribuído com parte do conhecimento que levei para o doutorado. Agradeço também ao professor Célio Albuquerque pelas aulas de Redes Sem Fio por ter aceitado participar da banca examinadora. Agradeço também ao professor Artur Ziviani por ter aceitado participar da banca examinadora.

Agradeço também a todos meus amigos do Rio, da Bahia, de Minas e de Petrópolis por estarem sempre presentes.

Resumo

O emprego da arquitetura *Content-Centric Networking* (CCN) em redes em malha sem fio tem se tornado um amplo campo de pesquisa. Um dos motivos é a possibilidade de combinar as vantagens de uso do *cache* em nós intermediários da rede com o baixo custo de implementação de redes em malha sem fio. A utilização de *caches* por nós intermediários pode aumentar a vazão em redes em malha sem fio. O motivo deste aumento de vazão se deve ao fato de que conteúdos populares e conteúdos perdidos podem ser requisitados aos nós mais próximos do nó consumidor, sem a necessidade de percorrer todo o caminho até o nó produtor, proporcionando uma redução na probabilidade de perda. No entanto é notória a necessidade de estudos frente aos novos desafios enfrentados na utilização de redes em malha sem fio orientadas a conteúdo. Um desses desafios é reduzir tempestade de *broadcast* de pacotes CCN, mais especificamente os pacotes de interesse usados para solicitar um dado conteúdo. Dependendo da taxa de requisição de conteúdos esta tempestade pode se tornar um problema limitante nas redes em malha sem fio baseadas na arquitetura CCN. A literatura apresenta algumas propostas de protocolos e mecanismos destinados a reduzir o problema de tempestade de *broadcast* por pacotes de interesse. Apesar das propostas existentes, é possível destacar que para cenários de múltiplos consumidores requisitando os mesmos conteúdos e cenários com múltiplos produtores, ainda existe a necessidade de propostas que reduzam os impactos negativos da tempestade de *broadcast* e aumentem a vazão da rede.

Esta tese propõe três mecanismos chamados: *Probabilistic Interest Forwarding* (PIF), *Retransmission-Counter-based Interest Forwarding* (ReCIF) e o ReCIF + PIF. O primeiro mecanismo define uma probabilidade de encaminhamento de pacotes de interesse. O segundo limita o número de pacotes de interesse com base em encaminhamentos prévios desses pacotes. O terceiro é uma abordagem híbrida que combina os critérios de encaminhamento dos dois mecanismos anteriores. O desempenho das redes em malha sem fio, baseadas na CCN é avaliado com os três mecanismos propostos e também com o mecanismo de encaminhamento padrão CCN e com outro protocolo utilizado na literatura, o *Listen First Broadcast Later* (LFBL). O desempenho das redes orientadas a conteúdo também é comparado com uma rede em malha sem fio com base na pilha TCP/IP executando o protocolo OLSR. Os resultados mostram que os mecanismos propostos fornecem uma taxa de entrega de conteúdo maior do que a fornecida pela rede TCP/IP com OLSR. Além disso, os mecanismos propostos superam o mecanismo de encaminhamento CCN padrão em até 22 % em termos de taxa de entrega de conteúdos em cenários densos com alto número de saltos entre a origem e o destino e proporcionam um atraso de entrega 49% menor do que a CCN padrão. Quando comparados com o LFBL, um dos mecanismos propostos, PIF, apresentou ganhos de 92% e 55% em termos de taxa de entrega e atraso respectivamente.

Palavras-chave: CCN, roteamento, redes sem-fio, CCWMN, redes de *cache*.

Abstract

The use of Content-Centric Networking architecture (CCN) in wireless mesh networks has become a broad field of research. One reason is the combination of cache deployment by intermediate nodes and the low implementation cost of wireless mesh networks. Caching by intermediate nodes can increase the throughput of WMNs. The reason for this increased performance is that popular contents can be retrieved from neighboring nodes to consumers. Thus, requests and contents do not have to traverse the entire consumer-producer path which reduces loss probability. However, the need for more research in this theme regarding the new challenges faced in the use of content-centric wireless mesh networks is notorious. One of these challenges is the broadcast storm problem. This broadcast storm is caused more specifically by the interest packets that request contents. Depending on the content request rate, the broadcast storm can become a limiting problem in wireless mesh networks based on the CCN architecture. The literature presents several proposals for protocols and mechanisms to reduce the problem of the broadcast storm caused by interest packets. Despite the existence of proposals, we emphasize that for multiple consumers requesting the same contents and scenarios with multiple producers, there is still a need for proposals that reduce negative impacts of the broadcast storm problem and increase the network throughput.

This thesis proposes three mechanisms: Probabilistic Interest Forwarding (PIF), Retransmission-Counter-based Interest Forwarding (ReCIF), and ReCIF + PIF. The first one defines a probability to forward interest packets. The second one limits the number of interest packets forwarded based on the number of previous forwarding actions of these packets. The third one is a hybrid approach that combines the forwarding criteria of the two previous mechanisms. The performance of a content-centric wireless mesh network is evaluated with the three proposed mechanisms and also with the default CCN forwarding mechanism and Listen First Broadcast Later (LFBL) protocol. The performance of such network is also compared with the OLSR protocol in a wireless mesh network based on the TCP/IP stack. Results show that the proposed mechanisms provide a higher delivery ratio than OLSR. Also, our proposals outperform the default forwarding mechanism by up to 22% regarding data delivery rate in dense scenarios with a high number of hops between source and destination and provide 49% lower delivery delay than the default CCN. One of our mechanisms, PIF, outperforms LFBL regarding data delivery rate and delivery delay by up to 92% and 55% respectively for high saturation levels.

Keywords: CCN, Routing, Wireless Networks, CCWMN.

List of Figures

List of Tables

Acronyms

APFS	:	Advanced Perceptive Forwarding Strategy
CBR	:	Constant Bit Rate
CCN	:	Content Centric Network
CDN	:	Content Distribution Networks
CM	:	Chunk Map
CP	:	Content Provider
CPR	:	Cost-Performance Ratio
CS	:	Content Store
DAS	:	Dynamic Adaptive Streamig
DASH	:	Dynamic Adaptive Streaming over HTTP
FIB	:	Forwarding Information Base
ICN	:	Information Centric Networking
ISP	:	Internet Service Provider
LFBL	:	Listen First Broadcast Later
LRU	:	Least Recently Used
LSA	:	Link State Advertisements
LSDB	:	Link State Database
MPR	:	Multipoint Relays
OLSR	:	Optimized Link State Routing
PIF	:	Probabilistic Interest Forwarding
PIT	:	Pending Interest Table
ReCIF	:	Retransmission-Counter-based Interest Forwarding;
STCR	:	Social Tie Based Content Retrieval
TC	:	Topology Control
WMN	:	Wireless Mesh Network

Contents

Chapter 1

Introduction

Wireless Mesh Networks (WMNs) are envisioned as a solution to provide low-cost access to the Internet [27]. Basically, WMNs employ a wireless backbone composed of stationary nodes to increase the connectivity compared with ad hoc networks. Client nodes, mobile or not, access the Internet through one or more than one of these backbone nodes because they are usually configured as gateways. For this reason, nodes that are often in the shortest paths toward gateways forward more packets than the others do and thus consume more their resources [102]. In addition to the high concentration of traffic near gateways, WMNs also suffer from interference and collisions that are characteristics of the wireless transmission. Nodes contend to access the medium at each hop because WMNs are based on multihop communication. Therefore, packets that already traversed several hops can be discarded near to its destination, which leads to resource waste [78].

The information-centric networking (ICN) paradigm is a solution for the problems cited above in WMNs [12, 76, 34]. Information-Centric Wireless Mesh Networks focus on the content delivery regardless of the content physical location. The content requests and the contents themselves are forwarded based on names. Therefore, nodes have no addresses, and a content request is not necessarily addressed to a particular node with a well-known address. The ICN paradigm also assumes that all network nodes can cache contents [61, 18, 83]. Thus, any node can send the content in response to a request if the desired content is stored in its cache [28]. Consequently, a content request does not necessarily reach the gateway to retrieve content from the Internet, if another node has previously requested the same content. In this case, the traffic load in nodes near to the gateway decreases as well as the number of hops traversed by the content.

There are several architectures proposed in the literature to deploy the ICN paradigm both in the Internet and in wireless networks [54, 71]. Content-Centric Networking [48] is one of these architectures. CCN radically changes the TCP/IP paradigm and becomes more suitable to handle the challenges faced by new network applications [39]. Currently, users of applications, such as BitTorrent and NetFlix, are more interested in the content itself than who provides this content [22]. Today 52.88% of the Internet traffic is generated by Netflix, YouTube, and BitTorrent applications [1]. The way how consumers request contents in CCN improves data delivery rate when compared with other destination-oriented network architecture [48, 55, 57, 42, 6, 38]. CCN nodes request a content by sending an interest packet that is forwarded through the network until it reaches a node able to send back [1] a data packet with the desired content. With CCN, intermediate nodes can cache contents. This approach reduces the bottleneck on content producers path. That is the reason why CCN improves data delivery rate. CCN also has native mobility support, since consumers request contents based on names and not based on destination address as in IP networks. Also, nodes do not change their addresses during changes of access points [13]. Another feature of CCN is the reduction of control message overhead on the network. This reduction occurs because nodes do not need to propagate routing tables and link changes to their neighbors. These features classify CCN as a promising network architecture for the future networks [59], including wireless mesh networks [17, 16].

CCN-based wireless mesh networks employ in-network caching, and thus reduces traffic concentration near gateways and the waste of resources caused by the late drop of packets. Also, CCN-based WMNs can benefit from the broadcast nature of the wireless medium to disseminate interest packets. Hence, when a node sends a packet, more than one neighbor is able to listen this packet transmission. Neighbors are thus candidates to forward a request or content. These multiple forwarding candidates increase the probability of a packet to be delivered to producer nodes or to forwarding nodes, which are temporarily able to provide content from their cache.

CCN has also been considered a solution to increase throughput on cellular networks. Zeydan *et al.* [106] state that besides the huge cellular network evolution from 2G to 4G, the backhaul connections of cellular networks did not evolve at the same speed and thus are not following the mobile backhaul intra-traffic future needs. Several studies identify that the major driver of the backhaul problem in cellular networks is the video-on-demand traffic [74]. Access to videos, in this case, is asynchronous and depends on the video popularity [106]. Therefore, a way to improve the video transmission in cellular networks is to reuse cached contents by user devices [49, 82]. Thus, CCN is envisioned as a solution to reduce the backhaul load on cellular networks because forwarding nodes

[1] The content producer or a node that stores the content in its cache.

employ cache natively.

1.1 Motivation

The development of WMNs based on CCN, however, is a challenge because of the broadcast storm problem [11] caused by the forwarding of interest packets. Each node broadcasts an interest packet received for the first time. Also, every neighbor will repeat the same behavior leading to a storm of interest packets. The main reason for that behavior lies on the fact that every node has one interface. Therefore, countermeasures applied to wired CCN to control interest flooding will not be applicable on WMNs. Furthermore, there is no hierarchy between CCN nodes, which increases the contention for the wireless medium. In this context, nodes can experience a high collision rate depending on the number of consumers nodes. Another issue that increases the collision rate is the transmission rate of interest packets. The consequence of these collisions is the increase of packet losses and delivery delay. In this scenario with high collision level, more interest packets will be retransmitted increasing medium contention and thus, increasing delivery delay. This scenario characterizes the broadcast storm problem.

There are works that aim at reducing the broadcast storm problem, i.e., as the work proposed by Meisel *et al.* [72] and Kim *et al.* [54], but there are remaining problems concerning the strategies to reduce medium contention by interest packets. These problems are related to interests packet filtering combined with the number of nodes in the network. Concerning filter interests in the network, it is necessary, for example, to identify whether there will be any criteria for filtering interests. Part of these filtering strategies uses a combination of IP networks and CCN. Hence, these strategies are based on adaptions to use CCN over the IP layer [85]. Another type of strategy to filter interest is based on the distance of possible forwarding candidates nodes and the producer nodes. Therefore, this strategy restricts the ability to propagate interest only to nodes that are selected as part of the best route towards a content producer node [72, 54].

These aforementioned interest filtering strategies for content-based WMNs can be suitable when the relationship among consumers and producers obey a limit. However, if we increase the number of consumer nodes on a network, these filtering restrictions may reduce the performance of content delivery. This behavior occurs because these restrictions may reduce the ability of consumer nodes to obtain contents from adjacent nodes. Another problem with these restrictions happens when the network has more than one producer. In that case, restricting the ability to forward specific interests to few candidates nodes may lead to unnecessary bottlenecks near producers.

These challenges related to broadcast storm handling in CCN-based WMN motivate the studies of this thesis and consequently inspired the proposal of mechanisms focused on flood reduction.

1.2 Contributions

This thesis proposes a set of mechanisms that filter the forwarding of interest packets to reduce the adverse effects of the broadcast storm in WMNs. The goal of the proposed mechanisms is to decrease the number of interest packets forwarded and thus reduce the probability of collisions caused by these packets. Nodes, in this case, experience a higher delivery rate and less delay if the number of collisions decreases.

The first proposed mechanism is called Probabilistic Interest Forwarding (PIF). With PIF, high-centrality nodes forward packets with probability p. Therefore, PIF reduces interest packet flooding on central nodes by reducing the number of packets that a central node may forward. With the second one, called Retransmission-Counter-based Interest Forwarding (ReCIF), nodes use the number of hops traversed by an interest packet to forward or not this packet. ReCIF also has two operation modes, hard and soft. Both are used to define the retransmission threshold. This threshold is used to limit the maximum number of forwarding hops that a packet may travel towards a producer node. The third proposed mechanism combines both criteria of PIF and ReCIF and is referred to as ReCIF + PIF. The idea is to guarantee that interest packets forwarded fewer times have a priority to be forward by a node and other interest packets are forwarded with probability p. Our mechanisms are compared with both the default CCN forwarding mechanism and the Listen First Broadcast Later (LFBL) [72] protocol. We also compare a CCN-based WMN with a TCP/IP-based network running the Optimized Link State Routing (OLSR) protocol.

The content delivery rate, the average delivery delay, and the number of interests transmitted by contents received are the metrics considered in the simulation. Results show the proposed mechanisms are more efficient in scenarios with a higher number of hops between source and destination. For these scenarios, our mechanisms outperform the default forwarding mechanism by up to 21% regarding data delivery rate in dense scenarios with a high number of hops between source and destination and provides 25%

lower delivery delay than the default CCN. One of our mechanisms, PIF, outperforms LFBL regarding data delivery rate and delivery delay by up to 92% and 55% respectively for high saturation levels.

1.3 Thesis Structure

The next chapters are organized as follows:

Chapter 2 presents a brief overview of both wireless mesh networks and CCN architecture. It also discusses forwarding strategies to reduce broadcast storm on WMNs. Projects and scenarios are also discussed to highlight the importance of these networks and their applications. Section 2.3 presents studies to demonstrate that the use of cache in WMNs increases packet delivery rate. This section also argues that the CCN architecture is also suitable to reduce the deficiency, faced by wireless mesh networks, regarding bottlenecks and medium contention. Therefore, the CCN architecture and operation is presented, as well as the differences related to the use of CCN in wired networks and wireless networks. The challenges related to the broadcast storm are also presented and discussed.

Chapter 3 discusses several works related to the adoption of caching and ICN architectures in WMNs. Use cases and new quality requirements of CCN-based WMNs are also discussed. Proposals to deal with the broadcast storm problem in ad-hoc networks are also presented.

Chapter 4 introduces the three mechanisms proposed. We also explain how the mechanisms choose high centrality nodes that will restrict interest forwarding. PIF mechanism is detailed and presented by its algorithm. ReCIF mechanism and its variation are discussed and explained. There are also diagrams which exemplify PIF and ReCIF-Soft mechanisms in action. The idea is to elucidate how these mechanisms work and their interest restriction strategies. The junction of RECIF+ PIF is also present with its algorithm.

Chapter 5 details the simulation environment and the parameters used by the protocols. Characteristics of the OLSR protocol regarding routing and flood contention are explained. The LFBL protocol is detailed regarding the functionality and its form of broadcast storm restriction. This chapter also discusses the simulation results. We start our simulations comparing default CCN and the protocol based on the TCP/IP protocol stack, OLSR. The idea is to compare both strategies regarding content delivery and average delivery delay. After these comparisons, we initiate the comparative experiments with default CCN, LFBL, and our mechanisms. Experiments include the variation of content request rate, type of content request and number of consumer nodes. The variation of content request rate is conducted to evaluate the mechanisms behavior regarding network saturation. We also use two types of content request. First, consumer nodes request contents in a sequential way. Second, consumers request contents by popularity. We also increase the number of consumers on the network to evaluate the impact of multiple consumers requesting the same set of contents. Afterwards, we conduct experiments to compare the performance implications of cache size on LFBL and our mechanisms. We also conduct tests on a less saturated scenario, by reducing the number of nodes on the network. The idea is to check the performance of our mechanisms in a less populated scenario as well.

Chapter 6 presents concluding remarks and comments about future work.

Chapter 2

Wireless Mesh Networks and Content Centric Networks

Al-Arnaout *et al.* state that the cost to transfer data in wireless networks is higher than the cost to transfer data in wired networks [12]. The reasons for this higher cost are the contention for the channel and channel interference. These two factors contribute to reducing the network efficiency as the number of hops between source and destination increases [88, 63]. If intermediate nodes can cache contents, as CCN nodes do, the number of hops between consumers and producers can be reduced. Consequently, bandwidth is saved by avoiding retransmissions caused by losses on the wireless channel. Oh *et al.* [76] argue that is better to invest on storage devices that are getting cheaper by the day than to deal with bandwidth limitations and losses of wireless networks. CCN focuses on changing the protocol stack to make the network a more efficient content distribution infrastructure, without creating overlay approaches as done by Content Distribution Networks (CDNs) [67]. The main reason for that is because overlay approaches could inherit TCP/IP problems.

The next sections present an overview of Wireless Mesh Networks, then we present a resume of Content-Centric Networks, and thereafter introduce the CCN-Based Wireless Mesh Networks. The last section is focused on the broadcast storm problem and how it impacts the CCN-Based Wireless Mesh Networks.

2.1 Wireless Mesh Networks Overview

The main reasons why wireless mesh networks have attracted industry and researchers attention are reduced costs, ease of deployment and fast maintenance [100]. The cost is reduced when compared to other types of networks for a relatively broad geographic coverage. WMNs do not need cables as wired infrastructures do and thus are easy to deploy. This ease of deployment is an essential resource in areas where the use of cables becomes impractical or with high maintenance costs. Universities have been pioneers in WMN deployment to interconnect campuses buildings and provide Internet access to their students and professors [77, 21, 23]. Several urban projects also use mesh networks to provide a connection between great geographic distances endpoints and thus providing Internet for the community [CUWIN, 2008].

The main characteristic of mesh networks is the use of wireless routers, which are generally fixed and have an unlimited power supply. These routers have two functions: (i) forwarding traffic to the other routers that are part of the destination route and (ii) receiving and forwarding traffic to the client nodes, which can be connected through a wired structure or using wireless communication. Figure 2.1 illustrates a typical example of a WMN interconnected to an Internet Service Provider. Connections between the wireless routers form a mesh, as can be observed in the center of this figure. These routers can be connected to wired networks, access points or even mobile users. Internet access can be provided to users connected to the backbone through gateways.

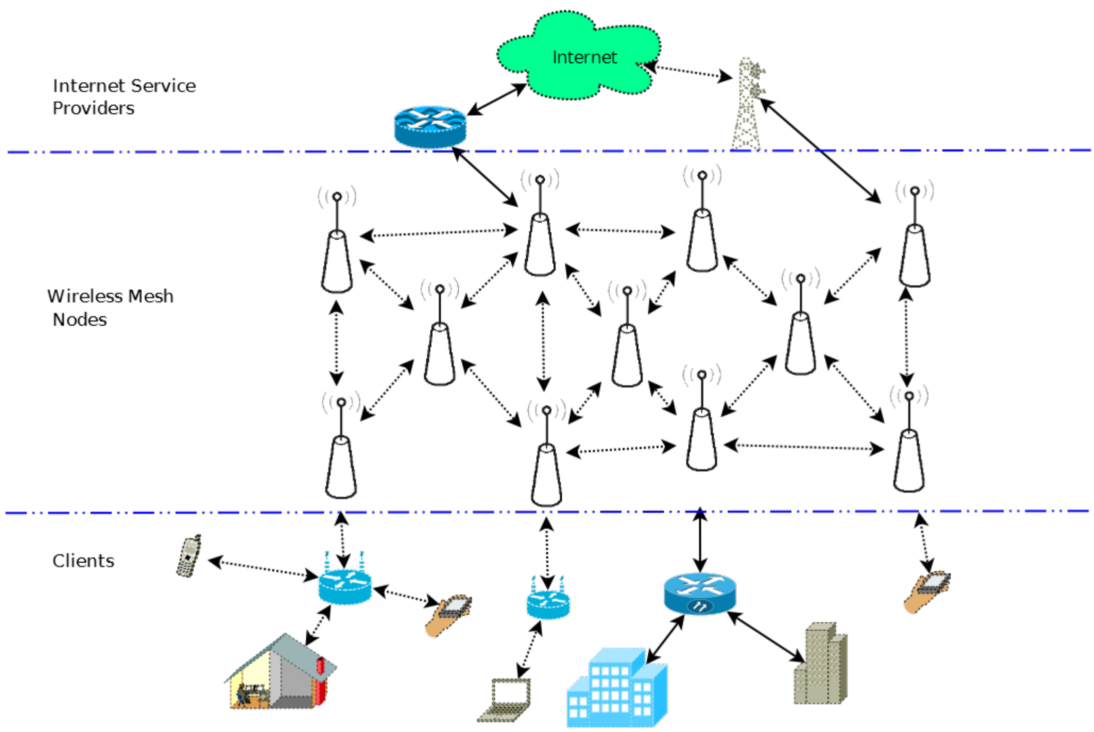


Figure 2.1: An example of Wireless Mesh Networks.

WMNs inherit self-organization from ad hoc networks. New nodes are added to network and routes are configured automatically [11, 90]. Therefore, auto-configuration makes the addition of new wireless routers easier and increases network scalability.

Routing is a keypoint in WMNs. Several studies propose to simply adopt routing protocols for ad hoc networks in WMNs [68, 86, 52]. The use of routing protocols for ad hoc networks, however, can degrade WMN performance, especially because these protocols have been developed for mobile networks. Wireless routers in WMNs are *quasi*-stationary. On the other hand, several proposed routing protocols for WMNs do not meet the general requirements of these networks. This problem occurs because many of the protocols have been developed to address applications with specific requirements such as high scalability, security, quality of service and military applications [56].

The behavior and performance of WMNs derive mainly from the combination of metrics and routing protocol[10]. Metrics take into account the packet loss rate, and others use multiple channels to provide better utilization of the physical medium, others also measure bandwidth. There are also metrics that take into account the number of nodes contending for the same physical medium and the interference caused by the nodes in their neighbors [103]. A considerable part of researches involving routing protocols and metrics for mesh networks are focused on increasing throughput[10, 40]. Hence, the importance of routing metrics for WMNs performance is related to the application type and topology used.

2.2 Content Centric Networks Overview

Content-Centric Networking (CCN) aims at changing the way how users request content, rather than requesting to machines the user would be able to request to the network [79]. This behavior implies in disrupting the packet forwarding based on the destination host and thus forwarding only by a content name. The structural behavior of CCN and its application have been studied in several works [25, 46, 48, 55]. Perino *et al.* [79] propose that the main features of CCN are (i) content items have an identifier address to be request instead of content hosts address; (ii) routing scheme takes into account content identifiers and not host identifiers for forwarding operations; (iii) content packets can be transparently cached by intermediary nodes along the path; (iv) native support for multicast routing, since a group of multiple consumers nodes can be satisfied by the same content. With regards to router equipment, CCN creates challenges. These challenges are related to hardware requirement to forward packets based on name and caching contents by intermediary routers. Therefore, changing from one billion IPs to approximately one trillion of content names may increase the routing state to be stored at content routers [79]. Conversely, using routers to storage contents can reduce the number of forwarding operations on the network.

CCN employs two types of packets: interests and data. Interest packets are sent by nodes to request content. These nodes are called consumers. Interest packets carry the name of the desired content instead of the address of a specific host/server. CCN uses hierarchical human-readable names to address content items [7]. These names are combined with several components delimited by a character, e.g., /UFF/NETWORK/PAPER1.pdf. Furthermore, routing in CCN can potentially increase the aggregation through this hierarchical naming scheme. Thus, when consumers request data, they specify what they are searching for and not where they expect it to be provided [14]. Data, or content, packets carry the requested content itself. These packets are sent by producer nodes or by intermediate nodes that store the desired content in its cache, referred to as Content Store (CS) and shown by Figure 2.2. In fact, a data packet would rarely carry the entire content. The content size is usually higher than the maximum data packet size [59]. Thus, content is divided into small pieces called chunks.

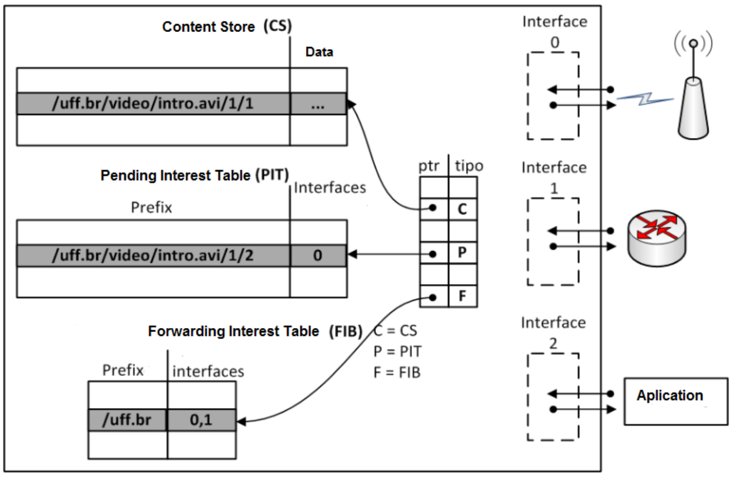
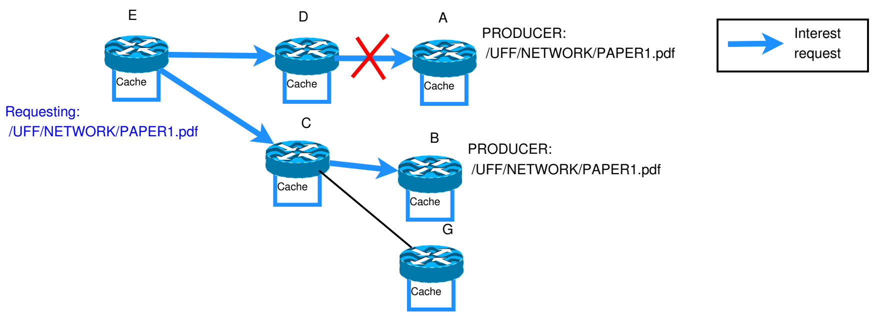


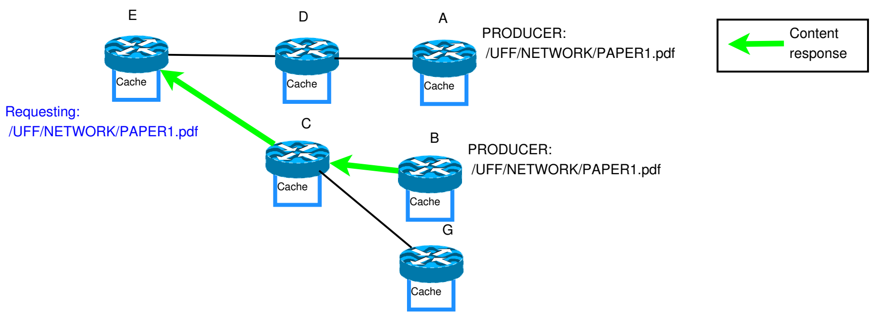
Figure 2.2: CCN node structure [35].

Two data structures are used during the forwarding of interest packets in CCN, as shown in Figure 2.2. The Pending Interest Table (PIT) keeps track of the state of each interest packet forwarded that did not receive a response yet, i.e., interests that are waiting for a data packet. Each PIT entry also records the list of receiving interfaces of interest packets for a given content. The second data structure called Forwarding Information Base (FIB) is used to forward interest packets to an output interface based on the content name. The FIB can be built at each node by the execution of a routing protocol that, similarly to the ones of current IP networks, would be used to exchange content name prefixes between CCN routers [43]. FIB maintains a list of entries containing name prefixes and output interfaces.

The process of forwarding interest packets is described as follows. As soon as a node receives an interest packet, it verifies its CS in order to find a copy of the content requested. If the content is in its CS, the node sends a data packet towards the consumer. This is the reason why authors consider CCN as cache networks [28]. Otherwise, the node verifies if there is another pending request for the same content on its PIT. If there is a PIT entry for the same content, the node updates the interface list of this entry and drops the interest packet. Otherwise, the node creates a new PIT entry and looks up its FIB to determine the output interface to forward the interest packet. If there is no FIB entry related to the content name, the interest packet is discarded. A node repeats this forwarding process for each interest packet received. Data packets follow the reverse path traversed by interest packets because PIT stores the list of receiving interfaces of an interest packet. Figure 2.3 shows an example of content requests. In that case node E starts to send interest packets to retrieve the the content /uff.br/video Figure 2.3(a). When the neighbors nodes D and C receive the interest, they forward the interest to other neighbors. Even if some interest did not reach the content, as i.e., node D to A, and exist another route to reach the content, it will reply with a content packet Figure 2.3(b).



(a) Consumer node E sending interest packet.



(b) Node B transmitting the content packet.

Figure 2.3: Forwarding on CCN.

2.3 CCN-Based Wireless Mesh Networks

As described in Chapter 1, CCN-based wireless mesh networks employ in-network caching and thus reduce traffic concentration near gateways and the waste of resources caused by the late drop of packets. Also, CCN-based WMNs can benefit from the broadcast nature of the wireless medium to disseminate interest packets. Hence, when a node sends a packet more than one neighbor is able to listen this packet transmission. Neighbors are thus forwarding candidates to forward a request or content. These multiple forwarding candidates increase the probability of a packet to be delivered to producer nodes or to forwarding nodes, which are temporarily able to provide content from its cache.

CCN-based WMNs typically are equipped with one interface only. Thus, the same interface is used to receive and forward interest packets. Therefore, this particular issue creates a higher forwarding spread possibilities since there would be no interfaces filtering on the CCN FIB table. Conversely, this behavior increases the interest broadcast storm since the routers nodes are unable to choose the proper forwarding interfaces. Therefore, upon receiving an interest packet for content that is not on its CS, the node has to forward the interest using the same interface where it was received.

CCN-based Wireless Mesh Networks are also suitable to reduce the deficiency faced by wireless mesh networks regarding mobility. That deficiency relies on most routing protocols developed for 802.11 [32] Ad-Hoc network is dependent on the level of physical mobility in the network to allow set up explicit routes between endpoints [72]. Thus those protocols require a level of track state for at least part of the topology and route configuration when links change. The reason for that relies on the fact that high dynamic networks impose different issues to IP addresses regarding assigned and reassigned to the constant changing set of active nodes in the network [80]. Depending on the application those address modifications also need to be mapped to tables with names and IP address. Comparing with CCN that problem would be reduced since the requests are no more destination oriented and thus content oriented.

Figure 2.3 exposes an example of CCN-Based Wireless Mesh Networks structure. The blue arrows express a prior request for content Y form the first consumer. After this request, the content packet follows the inverse path towards the consumer. Each CCN router of this delivery path, before forwarding the content, stores a copy of Y content. Thereafter, a new consumer requests the same content, Y, and its requests are represented by red arrows. We can observe that these request from the new consumer do not follow the path towards the Y content producer. Instead, the new requests get its response from an intermediary router. If those routers on Figure 2.3 were not caching contents, all the consumers request for Y content would be forwarded to the producer node. In this example, it is possible to see the importance of caching in reducing bottlenecks around paths that lead to gateways nodes, here called producers nodes. CCN nodes can store prior content and delivery that content when an interest packet arrives. Thus, reducing medium competition and interference traffic near producers nodes paths.

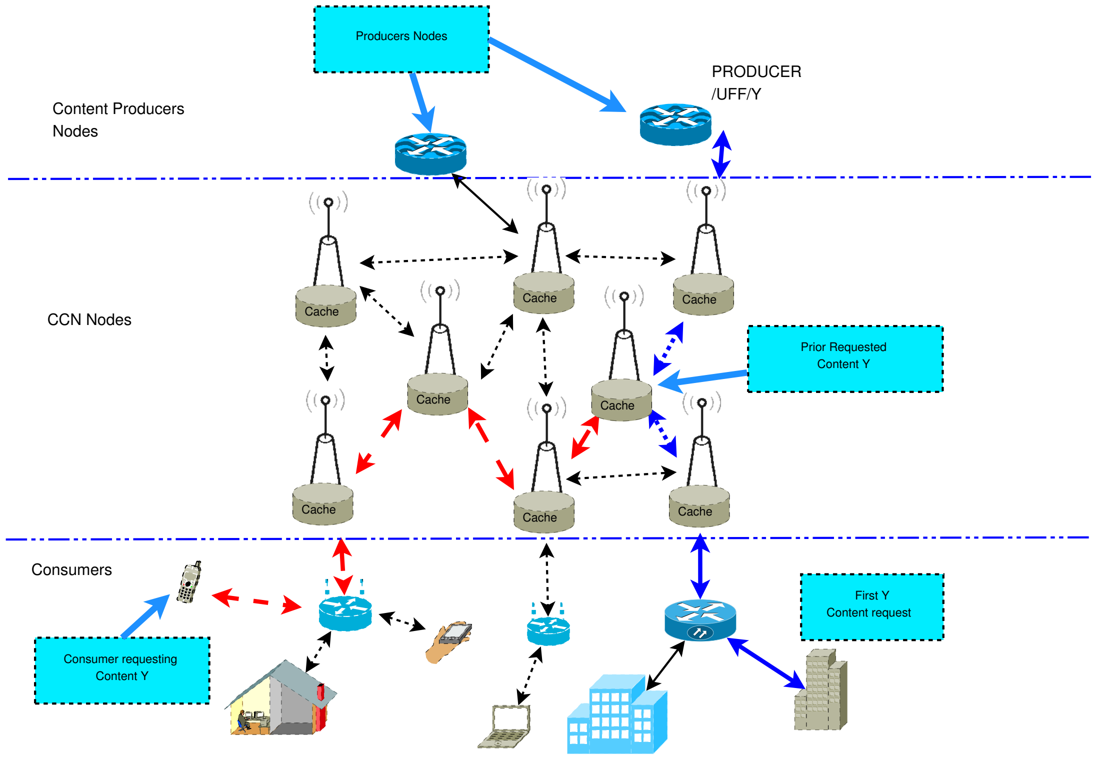


Figure 2.4: CCN-Based Wireless Mesh Networks.

Mobile Internet traffic demand has increased the use of wireless network nowadays, and the greater part of such traffic is content delivery, which is not real time in nature [63]. That traffic is suitable to be cached and inherits the possibility of content shared between consumers. The cache use provokes challenges on the ambit of how the storage resource will be managed. It has been reported that for both wired and wireless networks, caching consists of two main problems: content placement and content delivery [63]. For content placement issues like cache size, cache location and discarding policy are essential structures of the problem. The content delivery problem is related to how to use network resources efficiently to deliver content to their consumers.

Another important aspect of CCN is the capability to use caches to promote popular contents. It has been reported that users send more than 300,000 tweets, share more than 680,000 pieces of content on Facebook, and upload 100 hours of video on YouTube per minute [93]. However few contents induce popularity attention and concentrate most of the consumer requests [93]. Liu *et al.* [63] state that fundamental feature of content delivery traffic nowadays is that a minority of popular contents represents the majority of traffic load and is consumed by different users at different periods. Based on that behavior, it is important to predict if and when a content will become popular. Nevertheless, there is a shortage for predicting the popularity of a web content. That shortage is provoked by a sum of aspects that impact directly on content popularity for, i.e., content quality and the importance of a content for consumers [84]. Those aspects are difficult to scope and may vary over time. Tatar *et al.* [93] argue that combining information from different media providers is the first step to achieve a better prediction. The authors also indicate that Twitter has been used for this accomplishment, but there are other promising content providers.

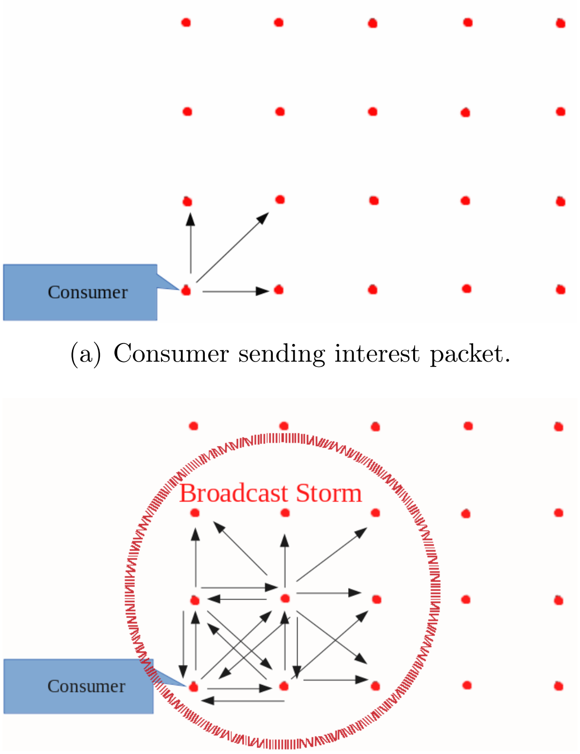
Regarding cache policy, there are two types: reactive and proactive. Reactive caching policy evaluates if a received content will be cached or discarded based on cache update. A simple way to use reactive caching policy is to store contents only after an interest packet has been sent. A proactive cache policy infers what content should be predictably cached before requests arrive. The use of proactive cache in CCN-based WMNs can improve the average throughput since nodes may take advantage of broadcast nature of the wireless medium to store contents that were not yet requested. Some proactive caching strategies use algorithms that predict what content should be cached based on consumers demand profile [63, 45, 19].

Liu *et al.* states that cache policy must take into account fading and interference on wireless NDN [63]. These problems are shown when a content is cached on distant neighbor nodes instead of the nearest neighbor of the consumer and thus the consumer request the content with lower signal power due to path loss and may also suffer from interference from the nearest neighbor node.

Content networks may also take the benefit of using a feedback related to acceptance for a specific content. Thus, consumers could work collaboratively with popularity prediction algorithms to proactive caches. To achieve that goal, consumers may vote on the content or manifest an opinion. Due to that, Tatar *et al.* state that one strategy to improve the popularity of a content is to account positive votes from consumers which have a conformist profile and also count votes from consumers which have a discontented profile [93]. Therefore, combining both groups of votes will enhance the popularity prediction.

2.4 The Broadcast Storm Problem

One of the main challenges faced by WMNs based on CCN is the broadcast storm problem caused by the way interest packets are forwarded. Depending on the interest transmission rate and the number of these packets forwarded, a network can experience a storm of interest packets sent in broadcast. Furthermore, data packets can be sent in response by neighbors at a high rate, which increases the medium access time and the number of collisions. These collisions are caused by nodes that are on the same path used to obtain the content or by nodes that are close enough to interfere each other [102]. Figure 2.5 demonstrates how the broadcast storm of interest packets starts. The consumer node starts to send the interest packet, as inherited behavior from the wireless medium is to receive information in broadcast, all near neighbors receive the interest packets Figure 2.5(a). Thus these neighbors will initiate the retransmission of the interest packets to their neighbors, and as a consequence, it will create a chain reaction where congestion and collisions will be more frequently, Figure 2.5(b).



(a) Consumer sending interest packet.

(b) Neighbors nodes retransmitting the interest packet.

Figure 2.5: Wireless CCN Broadcast Storm.

Concerning the broadcast storm problem, Tseng *et al.* [97] develop a mathematical model to study the broadcast storm on wireless networks. In this model, nodes coverage are modeled as circles, and the area of a circle determines the space covered by the wireless node signal. If two nodes are communicating with each other and the coverages are represented by a circle, then it is possible to determine the area where the signal, and as a consequence, the packets, from both nodes would be heard. This area is modeled as the intersection area of overlapping circles Figure 2.6. Thus, the authors argue that in a more realistic model, the unaffected area of a second node that receives packets sent in broadcast from its neighbor is 41%. It is worth mentioning that if more neighbors contend for the medium, the size of the unaffected area can decrease drastically. Tseng *et al.* [97] evaluates this impact through simulation. The authors randomly place n hosts on the same transmission range and calculate the unaffected area between those hosts. The unaffected area is below 5 % if $n \geq 4$. On the next chapter, we will discuss several mechanisms proposed to reduce the broadcast storm problem using cache on the wireless network.

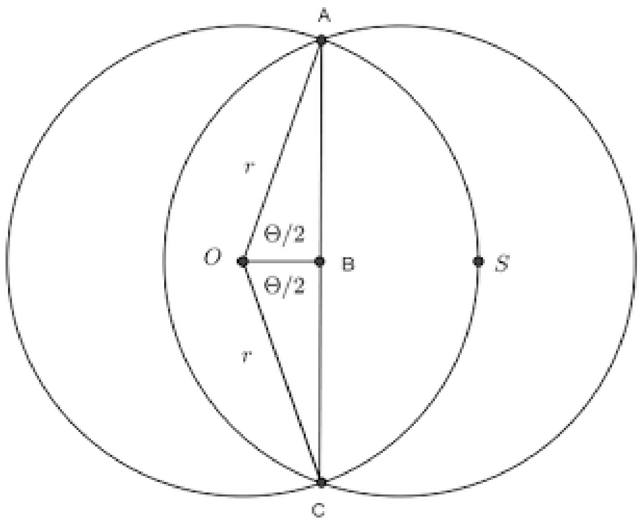


Figure 2.6: Intersection area.

There are techniques proposed for wired CCN to avoid the broadcast storm caused by interest packets. A simple technique is to define a rule per interface that states interest packets must not be forwarded to its receiving interface. This technique, however, is not suitable for CCN-based wireless networks because nodes typically are equipped with one interface only. Thus, the same interface is used to receive and forward interest packets.

Chapter 3

Related Work

The set of papers related to our work includes (i) mechanisms that employ cache in WMNs, (ii) proposals to replace the TCP/IP architecture by CCN in wireless networks, and (iii) novel CCN applications. Discussions about these four topics are presented in next sections.

3.1 Caching in Wireless Mesh Networks

Several authors propose to employ caching in particular nodes of WMNs. MP-DNA, for example, replicates contents to reduce the traffic overhead in WMNs [12]. Thus, if a node needs a retransmission of content or if any other node is interested in the same content, it can be sent by intermediate nodes. MP-DNA runs on top of the TCP/IP stack and considers that the OLSR routing protocol is running. Consequently, MP-DNA inherits the limitations of using TCP/IP in wireless environments. There are similar works to MP-DNA, such as MESHCHORD [24], that employ distributed hash tables to keep track of the content requested by neighbor nodes. In our work, we avoid the use of TCP/IP stack by adopting a stack that supports mobility natively.

3.1.1 CCN in Tactical and Emergency

Oh *et al.* [76] employ a CCN-based wireless network with mobile and stationary nodes. The authors, however, do not modify the CCN forwarding scheme to deal with the broadcast storm problem. They compare both the CCN stack and the TCP/IP stack + OLSR in a military environment. Results show that nodes with CCN experience a higher delivery rate and a lower average delay. Different from Oh *et al.*, we introduce new forwarding mechanisms to reduce the broadcast storm caused by interest packets.

3.2 Listen First Broadcast Later

The Listen First Broadcast Later (LFBL) protocol is classified as a content-oriented routing protocol. Therefore we choose this protocol to compare it with our mechanisms. Another point that motivated the study and comparison is related to the number of academic studies that mention LFBL [72, 47, 71, 15, 65, 104, 101, 17, 105, 16]. This way, our mechanisms are compared to LFBL and the default CCN in most of the analyses.

LFBL protocol differs from other routing protocols for wireless networks because it does not inherit protocol characteristics that were originally developed for wired networks. LFBL does not need predetermined routes and does not require the use of IP protocol, for example, LFBL is considered more robust to topological changes than destination-oriented protocols, such as OLSR and AODV. The reason is that data packets are not addressed to a particular node. Instead, data packets can be requested by any node capable of delivering the requested data.

With LFBL, routing decisions are made by the node that is receiving a packet and not by the node that sent the packet. This routing decisions is the key point of LFBL. After receiving a packet, a node considered as a potential router listens to the channel for a while to verify if there are other nodes more suitable than the node itself to be the packet forwarder. During the listening time interval, if there is no better candidate to forward the packet, the node itself will forward the packet.

The LFBL authors argue that the protocol does not maintain a state of connections. Therefore, nodes that are in the path of an end-to-end flow evaluate if they are capable of being forwarders based on their distances to the destination. This behavior removes from LFBL the need for knowledge of network topology.

LFBL presents three main features, which summarize its operation. First, there is no stored knowledge about neighbors, topology, and routes. By using broadcast communications, nodes that are in the listening perimeter can decide if they can forward the packet. Second, all the information related to routing is within the packet as LFBL only employs broadcast communications for sending packets. Finally, LFBL supports physical node mobility and logical mobility of data. The logical mobility comes from the fact that LFBL makes content oriented requests, and thus not to the destination as the IP protocol.

The end-to-end communication is divided into two phases: the request phase and the data phase. The request phase is similar to the default CCN forwarding, where requests are sent over the network without prior knowledge of destination location. During this phase, nodes simply forward the request until it is answered by a content-carrying node. The data phase begins when the request arrives at the node containing the requested content. In this case, the data is sent back using the candidate nodes for delivery. This phase continues until the requests cease, by receiving the content, or if there is no more content response.

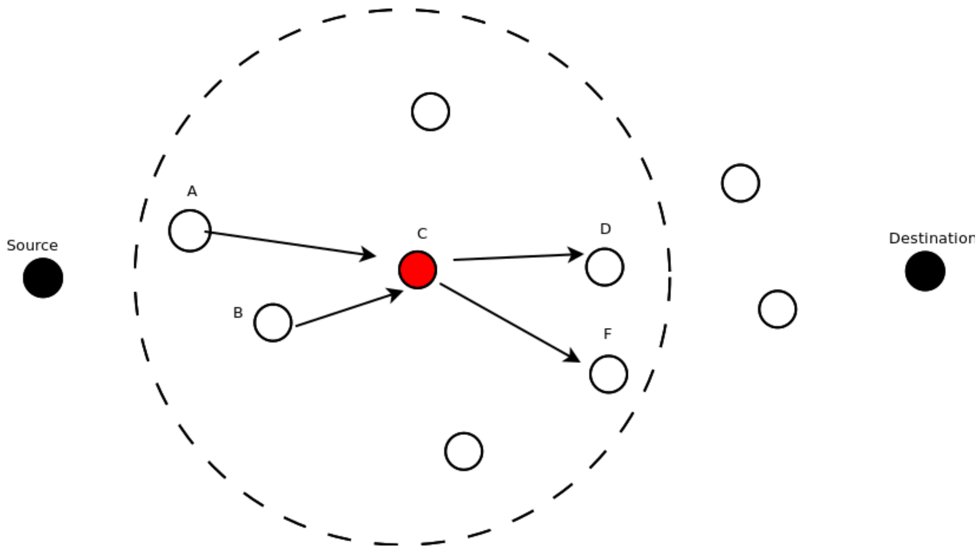


Figure 3.1: LFBL Forwarding Nodes Selection.

Figure 3.1 presents an example of forwarding nodes selection on LFBL. When a node C forwards the packet, a group of nodes within its cover range is able to listen that packet. Depict nodes A and B are able to listen the interest transmission, only nodes D and F are closer to the destination in number of hops. Therefore, only these nodes will be selected as possible forwarders.

LFBL inserts a header into the packet in order to have end-to-end flow information available. Table 3.1 summarizes all header fields. This header is used for the request and delivery phases. During these phases, nodes verify this flow header for packet forwarding decisions. Because routing decisions are performed by receiving nodes, it is important to note that a node must wait a given time interval before deciding whether it will be the best candidate to route the packet. During this time, nodes listen to messages broadcasted by other nodes. After receiving these messages, a node is able to analyze the LFBL header of these messages and verify if there are any forwarder candidates better than the node itself, i.e., nodes closer to the destination. Thus, a node receiving a packet must wait

Table 3.1: LFBL Header Fields

LFBL Fields	Description
seqnum	Sequence number created and added by the source node.
acknum	Acknowledgment number provided by source node.
srcId	Packet source node ID.
dstId	Packet destination node ID.
srcDist	The distance between the source node and the last intermediate node that sent the packet.
dstDist	The distance between the destination node and the last intermediate node that sent the packet.
type	Identifies whether it is a content request or response message.
dataName	The name of the content requested or answered.

a while and during that time count if there were better candidates and only if it still considers that there are no better candidates the node should forward the packet.

A node knows its distance to another node by using the srcDist field in the LFBL packet header, Table 5.1. This field stores the distance information between the packet source and the last node that forwarded the packet. At each hop, the node that forwards the packet updates its distance to the packet source into the srcDist field before sending the packet. Therefore, the srcDist field is used for nodes to discover and update their distances to source nodes.

Using the distance information between the current node and the source provides the ability to given node to use this information to compute if it will be among the candidates for the return flow to the source. Therefore, using the fact that a node can only be considered a candidate to the routing if it is closer to the destination.

The listening period is a key parameter of LFBL protocol. Assuming that in a wireless mesh network there would be more than one node transmitting its data, the listening time for a given node should be long enough to receive the neighbor packets and verify if it would be a forwarder or not. The authors argue that this time serves primarily to give priority to communications and also to avoid collisions. Priority refers to giving shorter time to nodes which are already chosen candidates for forwarding the packets. In this way, nodes that have previously forwarded packets from a particular flow would have priority over nodes that have not yet forwarded packets from this flow. The listening time is also used to reduce the collision because two forwarding nodes are not heard for the same period. By listening for a period, the same two nodes could consequently send the data exactly at the same time, as both would act as if they were the best candidate to forward the packet. Therefore, the authors argue that using a random factor to the listening period considerably improved the protocol performance.

Comparing with destination-oriented routing protocols it is worth mentioning that the closest point that the LFBL has in common with these protocols is the distance table. This table is made up of three fields and is used only by the node property, which means that this table is not passed on to neighbors. The table is being created as communications flow through the node. The node retrieves the information to populate its LFBL packet header table. The three fields of a distance table are the sequence number, distance to the destination and distance deviation. The sequence number is taken from the seqnum field of the LFLB header. This field is used on the table for making decisions about updating concerning how fresh the packet is. In this particular case, packets with an obsolete sequence number will not be used to update the table. The other field related to the distance to the destination is updated when the sequence number is greater than or equal to the current sequence number in the table. The third field, distance deviation, is calculated before the node includes its distance in the packet. The distance deviation will be calculated with the old value and the new one updated after receiving the packet. LFBL calculates distance deviation similarly to TCP calculates round-trip time deviation.

LFBL consumers create a local table to store its producer nodes. This table will be used after the flooding phase, i.e. when the nodes already know who the receivers are. Thus, a node can use this table to send interest directly to a producer instead of flooding the entire network using the request phase.

During the request phase, nodes learn who are the possible nodes that respond to requests for content. In addition to learning who the nodes are, it is also learned the relative distance of each node to the possible content server. The way back is done in the response data phase. At this stage, the content producer can analyze its distance to the node that originally requested through the dstDist field. This field is used by intermediary nodes to make the inverse path to the request and consequently to deliver the content to the node that requested it.

In fact, LFBL creates different flows between source and destination. These flows, after the initial request phase, are destination-oriented. Hence, when these flows are exposed to a densely populated network with a high degree of requests, create some bottlenecks in the vicinity of the producing nodes. This behavior is detailed in Chapter 6. We have also observed if multiple content servers are in the network the calculation of the dstDist field becomes imprecise. This imprecision is because for different responses of different content servers the intermediate nodes will assume their eligibility in a precise way and not necessarily reflecting the best routes.

The description of the LFBL does not detail information on how caching is done. For the sake of fairness compared to our mechanisms, we enabled the LFBL cache with the same sizes used by our mechanisms. LFBL also does not detail how long to wait during the request flooding phase. In our analysis, we realized this time severely impacts the performance of the protocol. For our simulations, we use 20 first seconds.

3.2.1 AIRDrop

Kim *et al.* propose a CCN-based WMN packet delivery scheme using unicast transmissions [54]. According to the authors, unicast transmissions reduce contention and eliminate additional network control messages. Therefore, once the path between source and destination is established, the packets are sent in unicast in this path. The name of the proposed scheme is AIRDrop, that name is because according to the authors, there is a similarity with the supplies distribution in a war environment and their scheme. Hence, if the supplies were thrown by airplanes the nearest soldiers on the field would be the aptest to collect the supplies.

Kim *et al.* argue that AIRDrop was designed to prevent packets from been lost at a given stage of the transmission process. That stage occurs when the hidden terminal problem affects the packet reception. They also point that on traditional mesh networks a packet on that situation would be summarily discarded. To overcome this problem, the authors argue that AIRDrop uses RTS/CTS MAC layer control transmissions to perform the retransmissions and avoid the hidden terminal problem. The authors argue that by using unicast with RTS/CTS the results outperform similar works that use broadcast transmission.

AIRDrop authors state that their work presents as a novelty the fact that it handles the path disruption taking advantage of the CCN resources. This affirmation relies on their assumption that since packet copies are stored in each retransmission node, this may be used to answer the future request for the same content. Hence, these copies would be explored to create a bridge with the broken unicast path. In the initial AIRDrop phase, the client node initiates communication by flooding a packet of interest to the first piece of content, and once it receives this content, it creates a bridge between the client and the producer. During this phase, other network nodes are aware of their distances between consumer and producer as in LFBL protocol. AIRDrop also inherits packet header control fields like those used on LFBL protocol. The intermediate nodes use these fields, which report network distances between source and destination, to make decisions about packet propagation.

Despite what is demonstrated in the proposed scheme, there are disagreements about this unicast approach. One of the reasons for the disagreement according to ETALL [29, 30, 109] is that the RTS and CTS frames are sent at the basic link bit rate, which delays the transmission of data frames that are sent in link bit rates near the maximum theoretical bandwidth. Ligo *et al.* argue that each RTS and CTS frame has a fixed downtime period and this also reduces throughput [62]. Consequently, this throughput reduction can exceed any gains to avoid hidden terminal collisions problem [62]. Another important fact presented in AIRDrop tests is that consumers only request different contents from each other. Consequently, the cache is partially used within the same message flow. Therefore, the gain related to prior content storage on intermediate nodes is not well studied.

3.3 Replacing the TCP/IP Architecture by CCN in Wireless Networks

3.3.1 CHANET

Amadeo *et al.* [16] propose a content-centric mechanism to act upon the IEEE 802.11 link layer. This mechanism, called CHANET, uses the 802.11 broadcast feature to provide content-based routing services. The mechanism disrupts the concept of routing based on destination address inherits from the TCP/IP protocol stack. According to the authors, CCN is suitable for ad-hoc wireless environments mainly because it improves the delivery rate in high mobility scenarios. The authors also explain that their mechanisms use exclusively broadcast packets, improving the simplicity, availability and robustness. Also, CHANET is compared with a TCP/IP stack network using AODV [80] routing protocol. Results show CHANET increases the data delivery rate and also reduces the time to deliver the contents. Despite the fact that the authors only compare their mechanism with AODV, they argue that representative experiments show the better performance of CCN over Optimized Link State Routing Protocol (OLSR). Therefore, OLSR was not used on their comparison results.

3.3.2 CCN Strategies for Multihomed Mobile Terminals

Schneider *et al.* propose strategies to use multiple interfaces for mobile devices (LTE, Wifi, Ethernet) with CCN [89]. The first strategy is to use the interface that best fits for a specific application QoS requirement or even for a security requirement. The other strategy is to send multiple copies of the same interest packet through different interfaces to guarantee redundancy. The objective is to access data regardless of the battery and throughput costs. Although the authors propose strategies to use multiple interfaces, there is no strategy concerning broadcast storm control.

3.3.3 Social-Tie-based Content Retrieval (STCR)

Lu *et al.* propose a social-tie based content retrieval (STCR) [66] scheme used in delay-tolerant MANETs. The authors use K-means clustering algorithm to build the social level to forward messages. The work is focused on sparse MANETs. All the nodes have an identifier and can compute their neighbors ID and their centrality, and social capacity to deliver a message to the destination or to forward the message to a better forwarding neighbor. The work differs from ours mainly because it is proposed to sparse networks and the nodes exchange information of the social hierarchy among them before requesting content. Also, sparse scenarios are less prone to suffer from the broadcast storm problem.

3.3.4 CCN-based Vehicular Network

Wang *et al.* introduce a CCN-based vehicular network [98]. The authors propose a mechanism to prioritize interest packets generated by distant nodes. Thus, a node postpones the transmission of interest packets from closest nodes always when there is a contention. Nodes are equipped with GPS devices to calculate the geographical distance between them. Wang *et al.* do not evaluate the mobility impact on the network performance. In this work, our proposals reduce the collisions of interest packets, but we do not require a GPS device.

3.3.5 Caching at the Wireless Edge

The work of Liu *et al.* provides a deep study on strategies that use cache on base stations or end user devices instead of using on the wired edge [63]. The authors observe that the larger the cache size, the higher the cache hit probability and thus the less the contention on backhaul bandwidth. The authors also argue that on current cache literature most of the studies have not take into account the channel fading and interference on wireless networks. Our work is focused on improving the throughput by reducing medium contention and interference.

3.4 Novel CCN applications

3.4.1 Virtual Wireless NDN

The work of Rainer *et al.* proposes a testbed architecture providing a virtual wireless Named Data Networking (NDN) [1] over an IP network [85]. The virtualization of NDN is modeled using network application as Iptables and TC on Banana Pi Routers. In this work, all nodes receive a unique identifier which will be used in the forwarding strategies. The authors compare the testbed with ndnSIM and discover some issues related to the TCP/IP headers overhead on the virtual NDN. Despite the TCP/IP headers overhead and the complexity of the forwarding tables with nodes identifier, the work is a major step towards real implementations strategies.

3.4.2 Service-based Congestion Control Strategy

Zhang *et al.* propose a service-based congestion control on content-centric networks [107]. This mechanism verifies the interest packet queue to decide if the CCN router is under congestion and if so the mechanism reduces the number of forwarded interests based on the priority of the data to be retrieved. This work, however, differs from our work because it is focused on wired CCN.

3.4.3 Energy-Efficient Caching Strategy

Shi *et al.* propose an energy-efficient non-cooperative caching strategy and replacement algorithm of the cost-performance ratio (CPR) [92]. The authors compare an Internet Service Provider (ISP) and content provider (CP) on a platform of end-user demand. Simulation results show that the strategy can effectively reduce network overhead and delay, and obtain better performance. The work mainly differs from ours because it is focused on wired networks and does not take into account the broadcast storm problem.

[1] Named Data Networking (NDN) is based on Content-Centric Networking (CCN) architecture and originally used CCNx as its codebase.

3.4.4 Caching and Computing Oriented Small Cell Networks

The content-centric network has also been evaluated on cell networks to reduce backhaul load and improve throughput. The idea is to cache popular contents on base stations (BSs) and in some cases mobile devices. The work of Zhao *et al.* proposes a framework that reduces the complexity of cellular networks and boosts the throughput using content-centric mechanism [108]. This work also exposes that caching popular contents is well desired for cell users satisfaction. The authors also show that as a consequence of using content-centric the interference among the small cell networks was reduced and thus the cost of the implementation is also reduced.

3.4.5 Big data caching for networking

Another work that targets backhaul load problem on cellular networks focuses on strategies and propositions of an architecture for big data in mobile cellular networks using proactive caching [106]. The authors aim for a better structure for the upcoming 5G wireless network with an extensive level of data processing. The idea is to set caches on the edges and reuse popular contents and thus improve the throughput.

3.4.6 Video Distribution over CCN

Jmal *et al.* propose the use of CCN in videos distribution networks [50]. The authors argue that the use of caches in this type of network increases the stability in video transmissions [95, 53] . The work contributes by introducing an integrated system to bridge the SAND [94] and DASH-aware network element. Hence, the authors implemented a bandwidth calculation system between CCN nodes and video servers. In addition to the calculating bandwidth system, the authors make use of Traffic Shaping to treat the adaptation rates for specific conditions that are still buffered at the nodes. These implementations provide significant gain when used in Dynamic Adaptive Streaming over HTTP (DASH).

Several works compare the use of CCN in Dynamic Adaptive Streaming (DAS) [58, 8, 64]. These papers use the CCN caching concepts to improve the rate of data delivery. The work of Rainer *et al.* [84] stands out in this theme. In this work, the authors compare a network based on CCN with networks using MPEG-DAS on IP-based networks. The work proposes a theoretical upper bound for multi-path multimedia transmissions without and with a cache. The authors concluded that the use of the CCN cache provides a better adaptation to multimedia traffic with different encodings and also improve multi-paths use for multimedia content delivery. Posch *et al.* [81] state that the use of information-centric networks has been presented as a viable option for multimedia traffic. The reason, according to the authors, is that these networks work better with different clients requiring the same multimedia content, but with different encodings. In this way, the base content can be distributed to several clients using the caches inherited from CCNs.

3.4.7 Advanced Perceptive Forwarding Strategy

The work of Li *et al.* propose a forwarding mechanism, called Advanced Perceptive Forwarding Strategy (APFS), which prioritizes the delivery of temporary contents replica [59]. In this mechanism, a second structure called Chunk Map (CM) is used in conjunction with APFS. This structure has the function to locate and inform nodes that contain the closest contents of a given node. The use of APFS in conjunction with the CM provides the choice of content replicas closer to the requesting node, reducing content delivery delay. The authors also argue that APFS provides better use of caching because the adjacent nodes will have a higher probability of providing content to neighbors.

3.4.8 OSPF Based Routing Protocol for Named Data Networking

Wang *et al.* creates an extension of the OSPF protocol to provide name-based routing capability in NDN, called OSPF Based Routing Protocol for Named Data Networking (OSPFN) [99]. The protocol operates on OLSR structure by distributing name prefixes and calculating routes to name prefixes. Since this project testbed was conducted to be operational in a short period, OSPFN is an extension of OSPF, and by that, it inherits IP related issues.

On OSPFN each router in the system collects network link state information and build a Link State Database (LSDB), similar to OSPF. This LSDB is propagated through neighbor nodes by Link State Advertisements (LSA) packets. When the network is in a converged state, all the routers in the network have the same copy of the LSDB. Therefore, each router constructs a network topology from the LSDB and runs the Shortest Path First to calculate the path to each destination node on the network. OSPFN uses OSPF's Opaque LSAs or OLSA to announce name prefixes, which allows application-specific information to be advertised in the network.

Routes to name prefixes are created due to an integration of CCND (Content-Centric

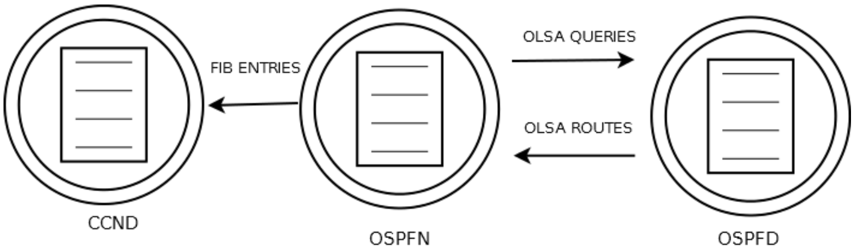


Figure 3.2: CCND, OSPFN and OSPFD Integration.

Network Daemon), which handles the forwarding of Interest and Data messages, and the OSPF. Therefore, every router uses CCND, OSPFN, and OSPFD to provide content oriented routes, as in Figure 3.1.

When a name prefix is advertised by multiple routers, something desirable on CCN, OSPFN sends a query for each producer nodes and inserts a FIB entry containing the name prefix and each next hop informed as a response. Although, OSPF protocol version used on OSFN provides only a single next hop for each destination node with different shortest paths. Thus, OSPFN by default generates one route for each content producer. Therefore, OSPFN generates a list of FIB entries for each name prefix. This list provides a rank where most preferred next hops are those that lead to producer nodes, ranked by path costs and followed by descending order of preference next hops.

The OSPFN project was released on Oct. 18, 2011, and is available for tests since then. The authors present an interest for a future work where OSPFN will not rely on IP addresses structure and thus, runs over a complete CCN structure. Although the protocol is not suitable for WMNs, the authors may conduct future tests in a near future.

3.4.9 Link Layer Broadcast Storm

Tseng *et al.* propose mechanisms to reduce the broadcast storm problem in the link layer [97]. One of their mechanisms is based on the probabilistic retransmission of IEEE 802.11 frames. In this work, we employ the probabilistic forwarding in an upper layer. We limit the forwarding of interest packets used by CCN.

3.4.10 Related Works Analyses

Besides the related works that aim at reducing the broadcast storm problem, there are remaining challenges concerning the strategies to reduce medium contention by interest packets. These problems are related to interests packet filtering combined with the number of nodes in the network and network topology. Therefore, it is necessary to choose the right criteria for filtering interests to avoid reduction on performance. Some related works present strategies to reduce broadcast storm from interest packets, i.e., LFBL, but when we introduce more producers nodes or increase the network saturation the performance decreases. Part of the related works uses a combination of IP networks and CCN. Hence, these strategies are based on adaptions to use CCN over the IP layer and therefore inherits IP related problems. Our mechanisms present robustness concerning network saturation and the number of producers. Hence, or mechanisms do not use combinations with IP, thus providing an entirely content-based forwarding.

Chapter 4

The Proposed Mechanisms

We propose three mechanisms to reduce the broadcast storm problem in CCN-based WMNs [69, 70]. The goal of our mechanisms is to reduce the number of interest packets forwarded by nodes and consequently increase network efficiency regarding delivery rate and delay. The following sections introduce the proposed mechanisms and discuss implementation details.

4.1 The PIF Mechanism

The first proposed mechanism is called Probabilistic Interest Forwarding (PIF). With PIF (Algorithm 1), a node forwards interest packets with probability p, if the node has a centrality degree $g \geq G$, where G is the centrality threshold. Otherwise, nodes that have a centrality degree $g < G$ forward interest packets according to the default CCN forwarding scheme. The centrality threshold G is defined based on the network density.

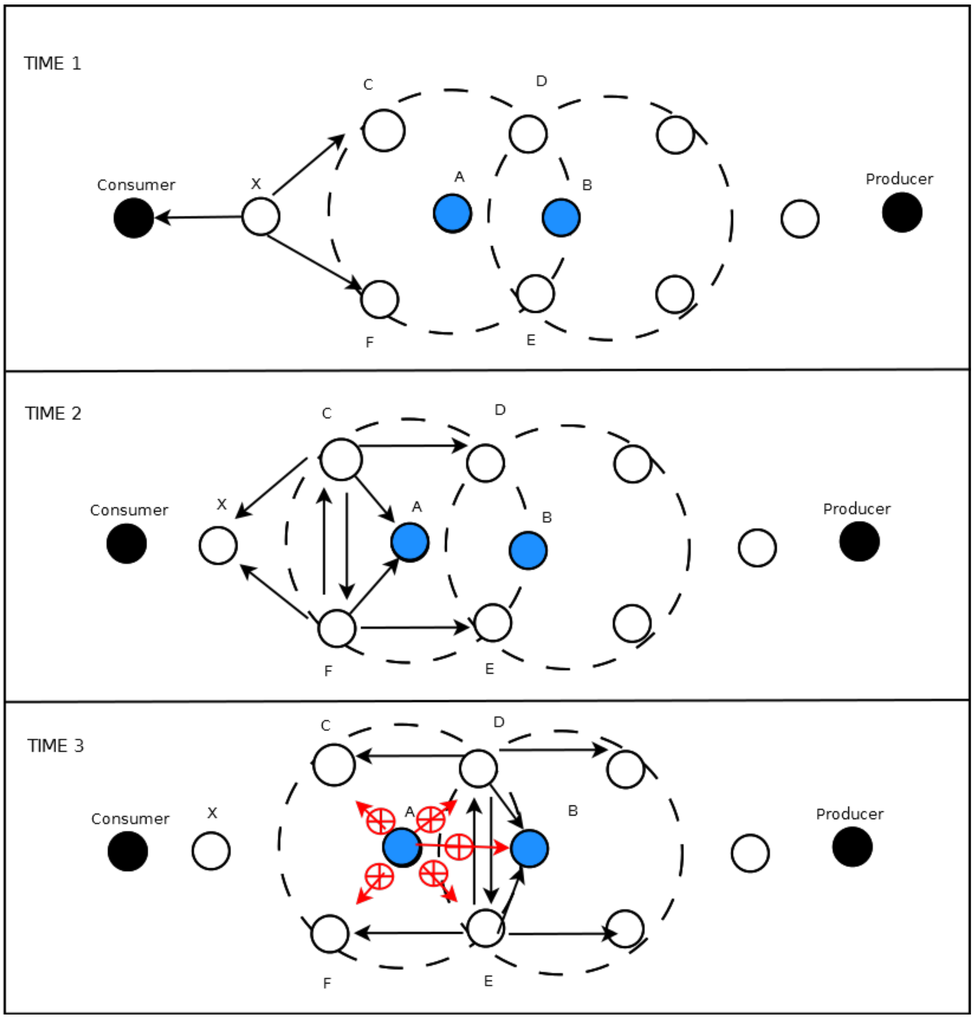


Figure 4.1: An example of PIF mechanism in action.

ALGORITHM 1: PIF Mechanism.

Data: (p, g,G, $pck.interest$.)

begin
  if $g >= G$ **then**
    $\lambda \leftarrow Random()$;
    if $\lambda =< p$ **then**
      $Forward(pck.interest)$
    else
      $Discard(pck.interest)$;
    end
  else
    $Forward(pck.interest)$
  end
end

With PIF, we aim at reducing the number of interest packets sent by nodes with high centrality degree, which means nodes with a high number of neighbors. Preliminary tests suggest the definition of a forwarding criterion based on centrality degree. We observed more nodes contending for the medium, just after an interest packet forwarding by a node with more neighbors. These nodes are trying to send copies of the same interest packet. Besides, a neighbor of a node with high centrality degree has a high probability of having already received that interest from another node, because interest packets are sent in broadcast.

Figure 4.1 presents a scenario where a consumer node requests a content to a given producer. Initially, node X will receive the interest packet and then broadcasts it. After this, node C and node F receive the interest and then broadcast to their neighbors. Assuming that node A is a central node, and there would be a probability p to forward this interest packet. If the node A choose not to forward the interest, there would be less medium contention. The reason for this is that if node A forwards this interest, it would affect nodes C,F,E,D and B. Therefore, filtering interest packets on central nodes provides a reduction on medium contention. This reduction on medium contention increases the network throughput because more interest and content packets would be able to be forwarded on that medium.

4.2 The ReCIF Mechanism

The Retransmission-Counter-based Interest Forwarding (ReCIF) mechanism introduces one field in the interest packet header. This field, referred to as forwarding counter, receives zero as an initial value and is increased by one unit each time the packet is forwarded. Nodes that have a degree centrality $g \geq G$, where G is the centrality threshold, forward an interest packet only if $c_i < C$, where c_i is the value of the forwarding counter in the header of packet i and C is the retransmission threshold. Otherwise, nodes that have $g < G$ forward interest packets based on the default CCN forwarding mechanism. ReCIF assumes that the forwarding counter of a packet increases fast during broadcast storm periods. Thus, ReCIF limits the forwarding of packets that were already in a broadcast storm. The fewer the number of copies of these packets, the lower the collision probability.

How to define the retransmission threshold C is a challenge to employ ReCIF. If C is low, interest packets may not reach its destination. Otherwise, if C is high, the efficiency of ReCIF is compromised. Therefore, we propose two operation modes for ReCIF: ReCIF-hard and ReCIF-soft. With ReCIF-hard, C is fixed and is previously defined based on network density (Algorithm 2). The threshold C is the same to all contents. On the other hand, ReCIF-soft calculates C dynamically (Algorithm 3). Thus, with ReCIF-soft, each content n has its own threshold C_n.

ALGORITHM 2: ReCIF-hard Mechanism.

```
Data: (g,G, C, c_i, pck.interest.)
begin
    if g >= G then
        if c_i <= C then
            Forward(pck.interest) ;
        else
            Discard(pck.interest);
        end
    else
        Forward(pck.interest)
    end
end
```

ReCIF-soft employs a two-phase algorithm to calculate threshold C_n. The first phase occurs when a node receives an interest packet for a specific content for the first time. To calculate the first value of C_n, a given node has to receive P interest packets. All these P interest packets are forwarded by the node following the CCN standard forwarding mechanism. After that, the node calculates the mean value of the forwarding counter c of these P packets and sets $C_n(0)$, as defined in Equation 4.1, We define $P = 10$ based on preliminary experiments.

$$C_n(0) = (\sum_{i=1}^{P} c_i)/P \tag{4.1}$$

With $C_n(0)$ defined, a node starts the second phase of the algorithm. To calculate C_n dynamically, ReCIF-soft employs an exponential moving average defined by Equation 4.2. The idea is to prioritize prior values of interests retransmissions and give a small weight to the recent ones. We use by default $\alpha = 0.9$ to adjust the threshold C_n for every interest received for the same content. That small weight can modify the threshold in a low scale. Therefore, high hop count values from few packets will produce a reduced impact on C.

$$C_n(t) = (((1 - \alpha) * c_i + \alpha * C_j(t - 1)), t > 0) \tag{4.2}$$

If a received interest packet of content n has c_i less or equal to C_n, then this packet will be forwarded, and the new C_n value will be computed. Otherwise, if c_i is higher than C_n, the packet will be dropped, but the new C_n will be computed. The reason to compute C_n from a discarded interest packet is to maintain the protocol awareness upon new network changes.

Figure 4.2 presents two scenarios where a central node B will decide if it forwards or discards a received packet. The blue nodes are the central nodes and are running the ReCIF-soft mechanism. Let us assume that the second scenario starts after the first scenario. For the first scenario, we assume that $C_n(t) = 5$, due to prior calculations. Thus, when the packet arrives at node B, after being forwarded four times, its c_i is 4. The node then compares c_i and $C_n(t)$ and then accepts to forward the packet, because the packet was forwarded less than five times. After this, B calculates the new value for $C_n(t)$ which is $(4 * 0.1 + 5 * 0.9) = 4.9$. Thereafter, for the second scenario the value of $C_n(t)$ is 4.9. The packet received by B in the second scenario went through 7 retransmissions, which is higher than $C_n(t)$. Therefore the packet will be discarded, and the new $C_n(t)$ will be recalculated to 5.11 thereon.

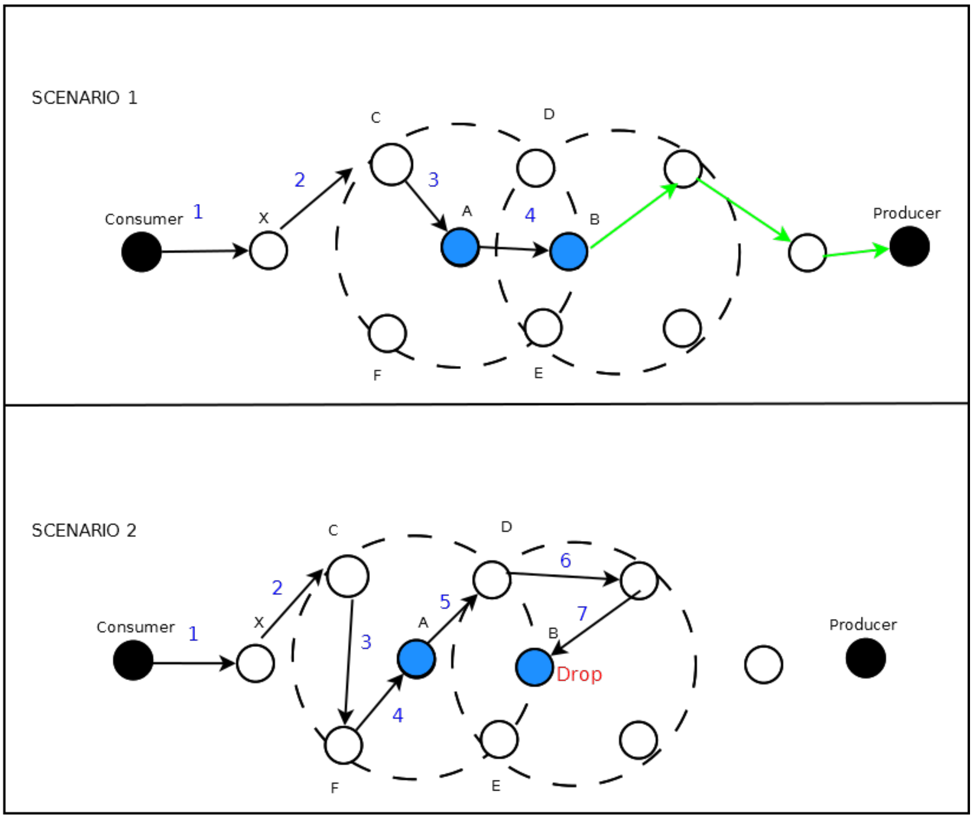


Figure 4.2: An example of ReCIF-soft in action.

ALGORITHM 3: ReCIF-soft Mechanism.

Data: (p, g,G, c_i, $pck.interest$.)

begin
    if $g >= G$ **then**
        if $ReceivedPcksN <= P$ **then**
            $sumCi \leftarrow c_i + sumCi$;
            if $ReceivedPcksN == P$ **then**
                $C_n(t) \leftarrow sumCi/P$
            end
            $Forward(pck.interest)$;
        else
            $C_n(t) \leftarrow ((1 - \alpha) * c_i + \alpha * C_j(t-1))$;
        end
        if $c_i <= C_n(t)$ **then**
            $Forward(pck.interest)$;
        else
            $Discard(pck.interest)$;
        end
    else
        $Forward(pck.interest)$
    end
end

We also evaluate ReCIF-soft with threshold $G = 0$ (Algorithm 4). Thus, all nodes forward interest based on the criterion adopted by ReCIF-soft, regardless of its centrality degree. The idea behind that strategy is to avoid the broadcast storm even on nodes located on low-density regions of the network. In some experimental traces, we have observed that nodes with low centrality degree also contribute to increasing the number of interests forwarded by nodes with high degree centrality.

ALGORITHM 4: ReCIF-soft G=0 Mechanism.

Data: (c_i, $pck.interest.$)

begin
    if $ReceivedPcksN <= P$ **then**
        $sumCi \leftarrow c_i + sumCi$;
        if $ReceivedPcksN == P$ **then**
            $C_n(t) \leftarrow sumCi/P$
        end
        $Forward(pck.interest)$;
    else
        $C_n(t) \leftarrow ((1-\alpha) * c_i + \alpha * C_j(t-1))$;
    end
    if $c_i <= C_n(t)$ **then**
        $Forward(pck.interest)$;
    else
        $Discard(pck.interest)$;
    end
end

4.3 The ReCIF+PIF Mechanism

The third proposed mechanism (Algorithm 5), called ReCIF+PIF, combines the features of PIF and ReCIF-hard. ReCIF+PIF also introduces the forwarding counter in the header of interest packets. Similarly, the counter receives zero as an initial value and is increased by one unit each time the packet is forwarded. Therefore, nodes with degree centrality $g \geq G$, where G is the centrality threshold, forward interest packets only if $c_i < C$, where c_i is the current value of the forwarding counter in the interest packet i, and C is the retransmission threshold. On the other hand, nodes that have a degree centrality $g \geq G$ forward interest packets that have $c_i > C$ with probability p. Thus, with ReCIF+PIF, how to define C is less critical because packets that are classified above this threshold still have a probability p to be forwarded. Nodes that have degree centrality $g < G$ forward interest packets according to the default CCN forwarding mechanism.

ALGORITHM 5: ReCIF + PIF Mechanism.

```
Data: (g,G, C, c_i, pck.interest.)
begin
    if g >= G then
        if c_i <= C then
            Forward(pck.interest) ;
        else
            λ ← Random();
            if λ =< p then
                Forward(pck.interest)
            else
                Discard(pck.interest);
            end
            ;
        end
    else
        Forward(pck.interest)
    end
end
```

4.4 Implementation Details

Table 4.1: New fields of the interest packet header.

Interest Packet Field	Description
FRW_counter	Incremented at each packet forwarding.
node_ID	Receives the identifier of the last forwarding node, modified at each hop.

With ReCIF and ReCIF+PIF, nodes need to know the number of hops traversed by a packet. Nodes running our three mechanisms also need to be aware of their centrality degree. For these purposes, we add two fields to the interest packet header shown in Figure 4.4: node_ID and FRW_counter. Table 4.1 summarizes the fields added to interest packet header. The FRW_counter field is calculated with each packet transmission. This way, each time the packet is forwarded, the FRW_counter field is incremented by one. The purpose of this field is to compute how many retransmissions a given packet has passed through.

The node_ID receives the identifier of the last node that forwarded the packet of interest. There are several proposals for neighbor discovery mechanisms in wireless networks [31, 68, 9]. For simplicity, we add a field to the interest packet header for this

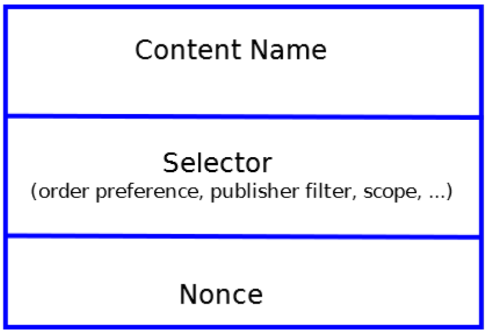


Figure 4.3: Default CCN interest packet.

purpose. Since the node only puts its identification when it is going to forward the interest, the other nodes that are receiving the interest can count the different node_ID to obtain an estimate of neighboring nodes. Despite the use of an identification field, this field is not used for purposes of identification of the origin or destination nodes. Figure 4.4 explains how a node obtains its centrality threshold:

1. After the interest packet arrival, the node checks its neighbor list size.

2. If the variable that stores the list of neighbors *neighbor.LIST* is empty, the node ID that transmitted the interest packet will be obtained from the arrived packet, and later this ID will be added to the list of neighbors. Together with the node identifier, the packet arrival time *pckt.TIME* is also stored. After that, the variable indicating the neighbors list size *neighbor.LIST.Size* is increased.

3. If the list of neighbors is not empty. The forwarded node ID obtained from the packet is used to compare with IDs stored on *neighbor.LIST*. If the ID is not found in the neighbor list, then the ID is added to the list (Step 2). However, if the ID is found on the list, then the packet arrival time is updated, *pckt.TIME*. The idea is to keep the neighbors on the list for a given time at least. Thus, we define a timeout for each entry. If the timeout expires, the tuple is removed from the list, and the list size is recalculated. The timeout is used to remove from the list nodes that may be off because of some failure or administrator change. The amount of time used to set the expiration time might change to cover specific network flexibility. For our work, that time was set to the time used in the simulation, since our nodes were fixed and the environment was controlled regarding node failures.

4. The last step is to check if any tuple of the neighbor list has its packet time higher than the expiration time. In that case, the tuple will be removed, and the neighbor list size will be decremented.

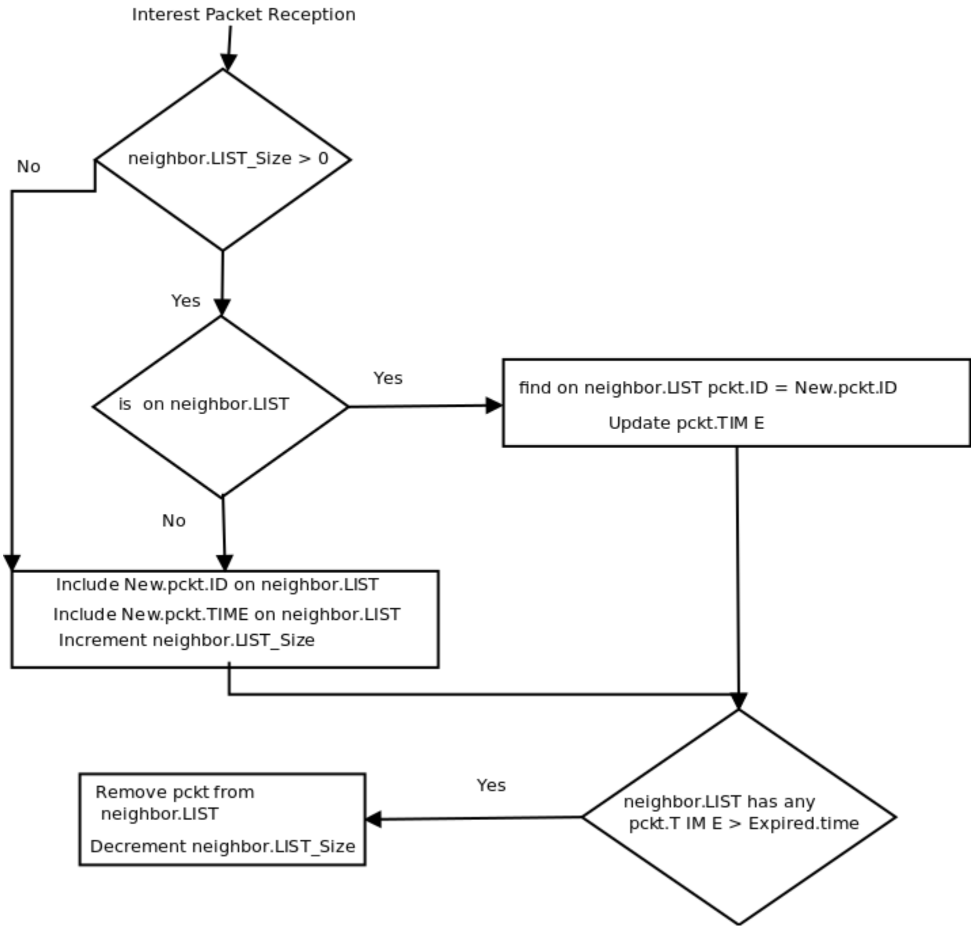


Figure 4.4: Neighborhood Calculation Flowchart.

Chapter 5

Simulation Environment and Results

We evaluate the proposed mechanisms through simulation by using the NDNsim module [5] of the NS-3 simulator. NDNsim implements the default CCN architecture. We have modified this module to add the support to wireless communications and our proposed mechanisms. In the following sections, we present simulation parameters and describes both protocols used in our evaluation: OLSR and LFBL. Then we present a comparison of OLSR and CCN. This comparison analyses the performance of both protocols regarding content delivery and content delivery delay. The other sections present comparing results of our mechanism and default CCN on different scenarios.

5.1 Parameters

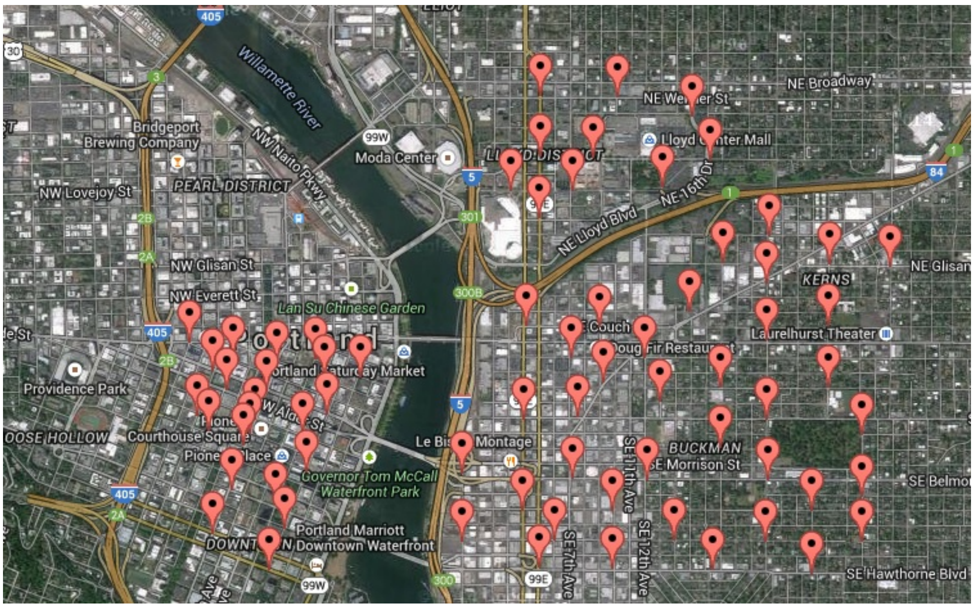


Figure 5.1: Geographical location of Portland network nodes.

Simulation considers a real network topology, running in an urban scenario. We use real traces from CRAWDAD [91] project that were acquired from a wireless mesh network operating in Portland, Oregon, USA, depicted by Figure 5.1. This network covers an outdoor scenario with urban buildings between some nodes, thus creating a set of routes which differs from a grid routes. The network consists of 70 nodes, and some of them are geographically separated by a river. The first reason for choosing that network is related to the number of nodes, which provides a dense concentration of nodes on some parts while in other parts of the network the concentration of nodes is barely sparse. That contrast of group of nodes with different densities provides an opportunity to evaluate our mechanism in a more realistic scenario. The producer nodes are fixed on the edge of the network, while the consumer nodes are randomly distributed. To simulate this scenario, which is urban and outdoor, the following parameters are defined. The network technology is the IEEE 802.11a standard in ad-hoc mode with OFDM (Orthogonal Frequency-Division Multiplexing) modulation and the transmission rate is 24 Mb/s. The propagation model used is the Shadowing Deviation with 5 dBm of transmission power.

We also vary the transmission rate of interest packets. Consumers send from 10 to 50 interest packets per second. Interest packets carry the required content name. Every content has a size of 1000 bytes and is encapsulated in one data packet. The cache size is equal to 10000 kB in all nodes and the cache replacement policy is the Least Recently Used (LRU). Depending on the scenario analyzed, content chunks are requested sequentially or based on its popularity defined by a Zipf-like distribution with parameter $\alpha = 0.7$.

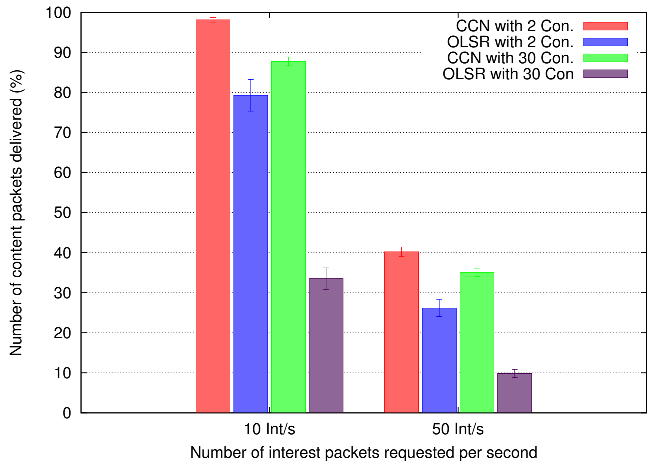
We have defined the following parameters for the proposed mechanisms. The centrality threshold G is equal to 8. We define $G = 8$ based on preliminary experiments. Both mechanisms, PIF and ReCIF+PIF, are evaluated for probability value: $p = 0.2$. The reason for that value is based on previous simulation results were $p = 0.2$ provides better performance. Finally, with ReCIF-hard, retransmissions threshold is $C = 7$ for the reality-based topology. These values are defined according to preliminary tests that indicate the average number of retransmissions needed to interest packets of a given consumer to reach the producer. ReCIF-Soft uses alpha = 0.9 and P=10.

Our experiments compare the performance of the proposed mechanisms – PIF, ReCIF and ReCIF+PIF – with both the default CCN forwarding mechanism and LFBL. We also conduct experiments to investigate the performance of CCN when compared with the TCP/IP stack running the OLSR protocol on a WMN-based scenario. The content delivery rate, the average delivery delay, and the number of interest transmitted by contents received are the metrics considered in the simulation. The rate of content packets delivered is calculated using the number of requested contents delivered on each consumer node. The content delivery delay is the time between the transmission of an interest packet and the reception of the corresponding data packet by a consumer. For each configuration described in Chapter 5, we perform 50 simulation runs of 100 s each. Confidence intervals are calculated for 95% confidence level and are represented by vertical bars in figures of the following sections. Results are split into interest request speed and number of consumers nodes.

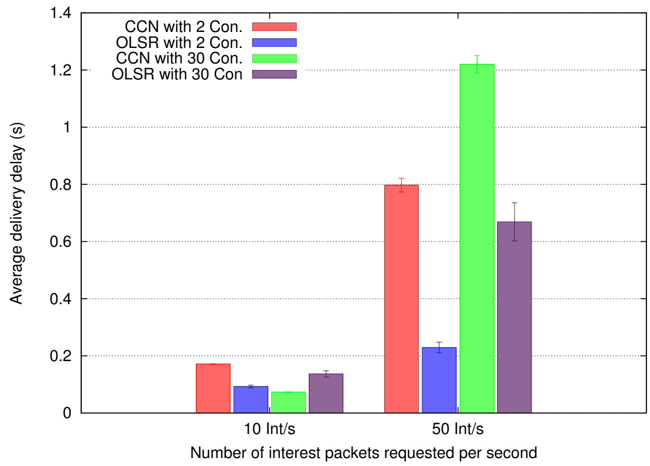
5.2 TCP/IP Stack with OLSR vs CCN

First, we compare the WMN based on the default CCN with a WMN based on the TCP/IP stack running the OLSR protocol. The main objective of this experiment is to investigate if CCN outperforms TCP/IP stack in mesh networks. The parameters of OLSR are the following. The interval between HELLO messages is equal to 1.5 s. Link state messages are sent on every 5 s. These messages are used by multipoint relays to calculate the topology map. We assume a traffic with Constant Bit Rate (CBR) to simulate the request and the delivery of contents in the TCP/IP+OLSR WMN. CBR packets have a size equal to CCN data packets. The goal of this configuration is to approximate the traffic behavior of the two networks because both networks operate in a very different way regarding data forwarding. Therefore, the CBR traffic is transmitted at the same rate of CCN consumers request contents.

Figure 5.2 shows the results for content delivery rate and average delay. We vary the number of consumers, 2 and 30, and the rate that each consumer sends content requests, 10 and 50 interests per second. The idea is to investigate the impact of the number of consumers and the rate of content requests in the two WMN deployment approaches. In Figure 6.1(a), vertical bars represent default CCN and TCP/IP+OLSR network for 2 or 30 consumers. Bars are grouped based on the rate of interests sent by consumers, 10 or 50 int/s. Results show default CCN delivers more content packets than TCP/IP+OLSR for all analyzed configurations. Furthermore, the performance gap between the two approaches increases as the number of consumers increases and the rate of content requests also increases. For example, CCN delivers almost 100% of content packets for two consumers sending 10/int/s. TCP/IP+OLSR delivers 80% in this configuration. However, if the number of consumers increases to 30, CCN delivers almost 90% of content



(a) The percentage of content packets delivered by TCP/IP stack with OLSR and by default CCN.



(b) The average delay of content packets delivered by TCP/IP stack with OLSR and by default CCN.

Figure 5.2: Default CCN vs. TCP/IP with OLSR: content delivery rate and average delay.

packets and TCP/IP+OLSR only 35%.

We also observe that the delivery rate severely decreases if the content request rate increases to 50 int/s. In this case, the network is saturated, and as a consequence, more packets are lost by medium contention and inter-flow interference. Default CCN performance is still higher than the one provided by TCP/IP+OLSR. This performance is a consequence of proactive caching employed by default CCN. With proactive caching, nodes store contents sent by other nodes in its transmission range even if there is no PIT entry for the content. Thus, content availability increases. With CCN, consumers retrieve contents from adjacent nodes and consequently interests, and data packets are forwarded fewer times. Thus, the medium contention is reduced. On the other hand, with

TCP/IP+OLSR, packets always traverse the source-destination path.

Figure 5.2(b) we observe for three of the four configurations that default CCN provides higher average delay than TCP/IP+OLSR. In addition, the higher the network saturation, the higher the average delay for both approaches. The reason for this behavior is directly linked to the possibility that more distant nodes might receive contents that were previously lost along the way using CCN caches. By computing the reception of these contents that have traveled a farther way, the average delay is also increased. Besides, the TCP/IP+OLSR packets were transmitted from the producers nodes to consumers nodes. The idea was to simulate content delivery after a successful request. Therefore, we compute only the content delivery delay for TCP/IP+OLSR without computing interest packet arrival delay.

Thus, we can conclude that CCN delivers more content than TCP/IP+OLSR. However, this increase, on content delivery rate, has a drawback related to a comparatively higher delivery delay. Network saturation is the key point to reduce the content delivery rate and increase the content delivery delay. Therefore, we propose mechanisms to reduce the number of interest packets transmitted and thus reduce the network saturation.

5.3 Our Mechanisms

We compare the performance of the proposed mechanisms PIF, ReCIF and ReCIF+PIF with both the default CCN forwarding mechanism and LFBL. We consider three metrics to evaluate the mechanisms: content delivery ratio, the average delay of delivered contents, and overhead. The first one is given by the ratio between the number of contents delivered and the number of interest packets sent. The delivery delay is measured for each content and is the time between the transmission of the interest packet and the successful reception of the requested content. Overhead stands for the number of interest packets forwarded for successfully received contents. The following sections discuss the results.

5.3.1 The Impact of Content Request Rate

First, we compare our mechanisms with the standard CCN and LFBL in a scenario with a high concentration of consumers with different content request rates. We consider 30 consumers requesting contents available in 2 producers, and we vary the content request rate from 10 to 50 int/s.

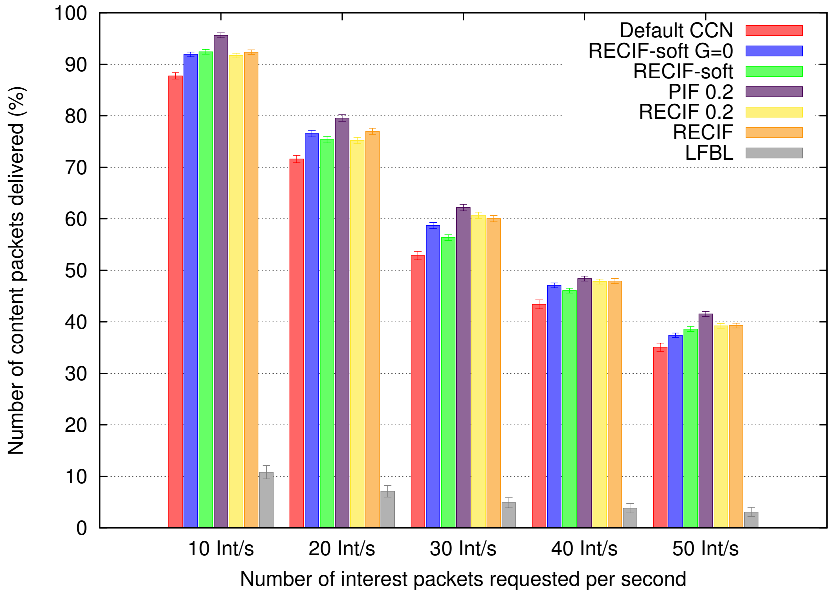
Figure 5.3 compare default CCN, LFBL and our mechanisms for different content request rates. Each consumer requests content sequentially. First, we conclude the proposed mechanisms outperform default CCN and LFBL regarding the number of content packets delivered, Figure 5.3(a). The proposed mechanism PIF with $p = 0.2$ (PIF02 for the sake of brevity) presents the highest performance as the content request rate increases. With PIF02, high-centrality nodes only forward 20% of interest packets received, as defined in Chapter 4. That strategy produces less broadcast of interest packets that reduces the competition for the wireless channel.

The medium competition also impacts the delivery delay Figure 5.3 (b). For all mechanisms, delivery delay increases as the content request rate increases, as expected. LFBL produces the longest delay compared to our mechanisms and the default CCN. The reason for this higher delay is related to LFBL characteristic of using independent flows for request and delivery of contents. This requirement of independent flows overloaded the regions closest to producers where a greater amount of interest and content packets travel. Thus, causing a higher delay in receiving content packets. The number of independent flows also reduces the number of interest packets delivered. The overhead of interest packets, Figure Figure 5.4, on the network center made nodes near to the destination to be overloaded. This overload increased the packets hop count by forcing part of the flows to use peripheral nodes to deliver content packets.

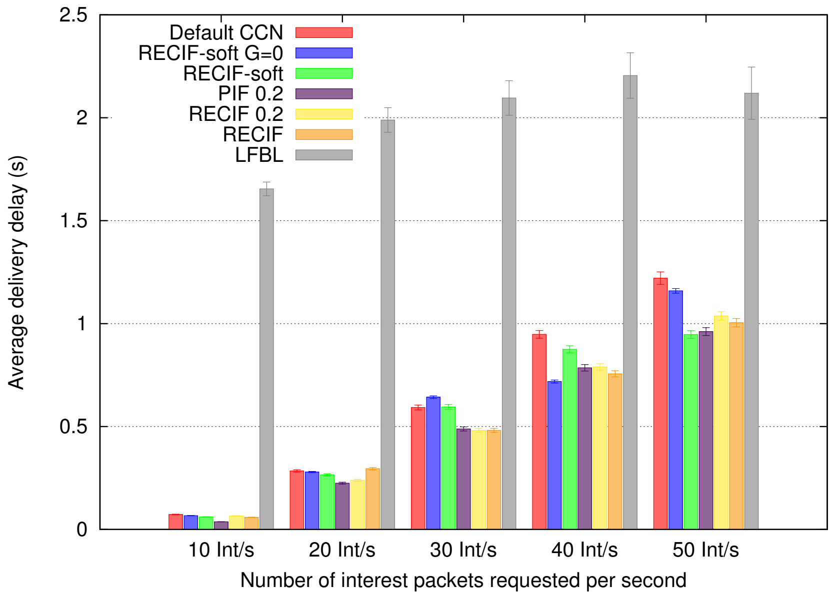
As the medium competition increases for higher content request rates the number of hops to deliver a content decrease. That is explained by the fact that, on more saturated scenario part of the interests packets from farthest consumers will not be able to arrive at their destination nodes with the shortest paths. That inability to fulfill the shortest path for some interest packets is mainly caused by the medium competition on the center of the network and in the region near the producer nodes. Therefore some interest packets from farther nodes reach their producers using more hops and with less probability.

The PIF02 mechanism delivers more contents than the other mechanisms. The reason for this behavior is related to the fact that the PIF, with its restrictive forwarding strategy, that more interest packets from far nodes could reach the producers. This delivery occurs because there is less saturation along the path and consequently more far nodes can receive more content packets.

We also change the way how consumers request contents. Now, consumers request contents according to a Zipf-like distribution to evaluate the mechanisms in a scenario where contents are retrieved from caches more often. Figures 5.5 and 5.6 show the results.



(a) Content delivery rate.



(b) Average delay.

Figure 5.3: Content packets delivered and average delivery delay for 30 consumers requesting sequential content at different rates.

Figure 5.5(a) shows that our mechanisms delivery more contents than default CCN and LFBL. Less restrictive mechanisms, such as ReCIF and ReCIF+PIF with $p = 0.2$ (RECIF02 for the sake of brevity), provide higher delivery rates. In addition, all mechanisms deliver more contents in this configuration than in the previous one (Figure 5.3(a)). This performance is a consequence of in-network caching, i.e., most popular contents are more likely to be obtained from adjacent nodes. With the increase of the content request rate, similar to the sequential request scenario, the nearest nodes of producers were more likely

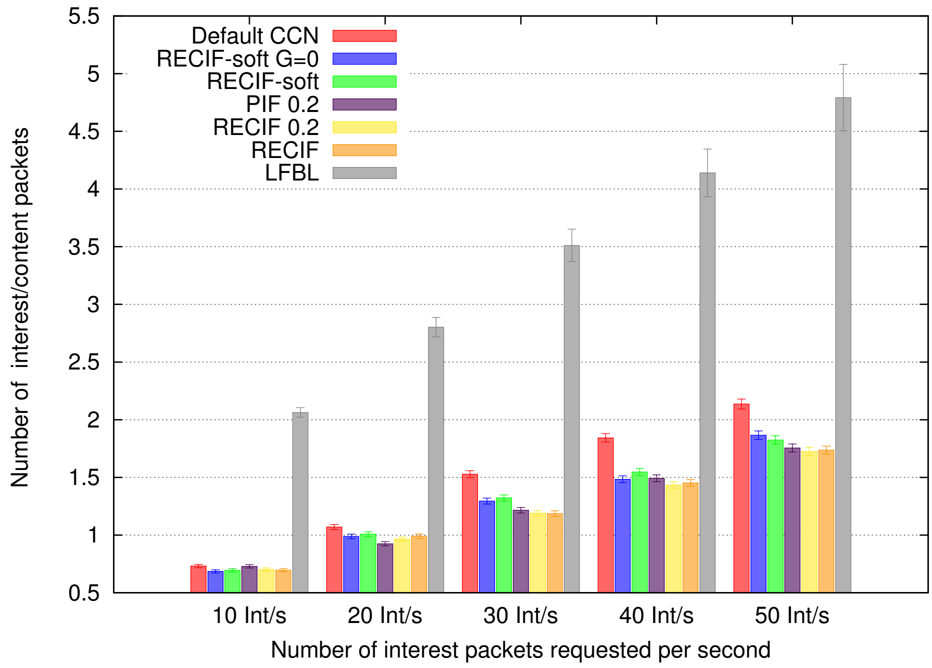


Figure 5.4: Overhead for 30 consumers requesting sequential content at different rates.

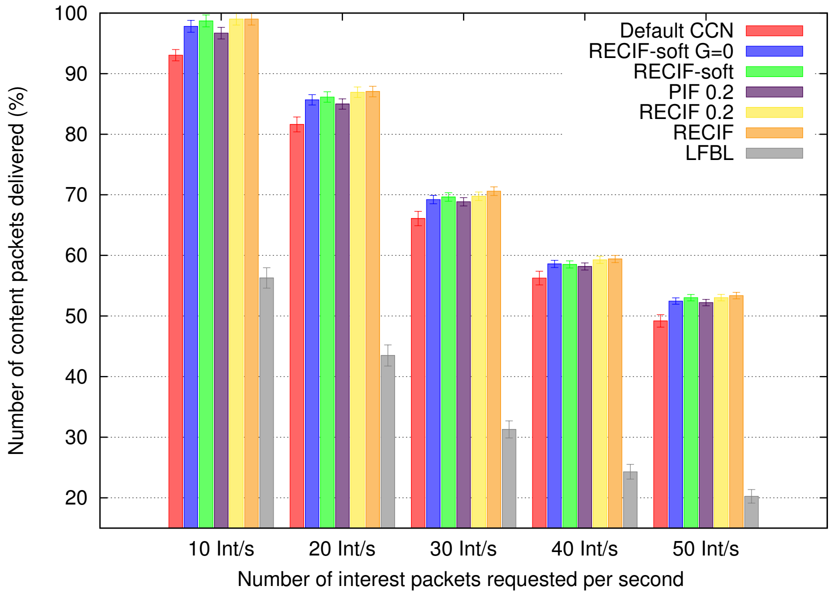
to receive contents.

Regarding LFBL, the delivery rate decreases as the content request rate increases. In the more saturated scenario, with 50 interests per second, the difference from LFBL to ReCIF, regarding content delivery, reaches 163.4% more contents using ReCIF. The reason for this behavior, similar to the one occurring in the sequential scenario, lies in the fact of multiple flows and consequently the LFBL's low capacity to obtain popular contents of the next nodes. The multiple flows and its impacts on overhead and average delivery delays can also be observed in Figure 5.5 (b) and Figure 5.6.

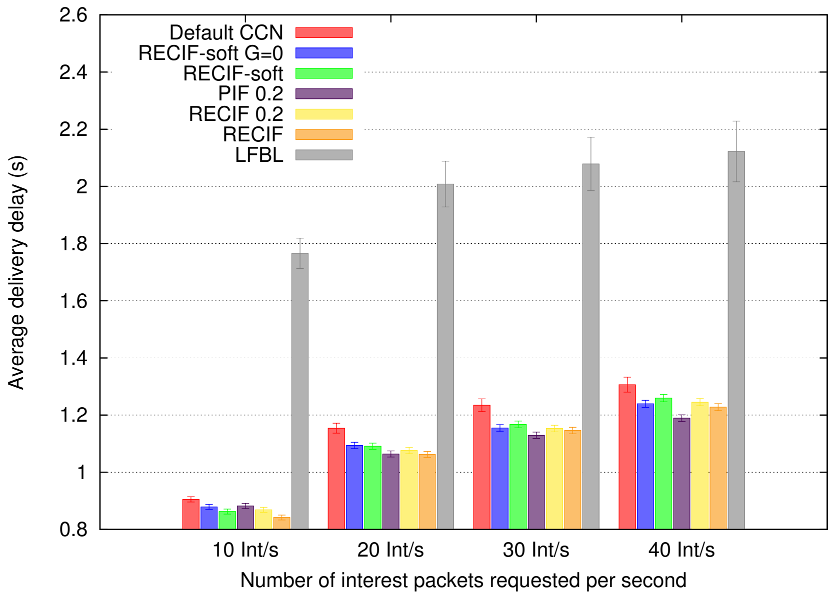
5.3.2 The Impact of the Number of Consumers

After varying the content request rate, we conduct experiments to evaluate the impact of the number of consumers on the network. The reason for that evaluation is to analyze the proposed mechanisms performance with different network saturation levels caused only by the increasing number of consumers. We set the content request rate at 50 int/s for each consumer. The goal is to observe if the network experiences high saturation level even for a low number of consumers.

First, consumers request contents sequentially, Figures 5.7 and 5.8. For two consumers, Figure 5.7(a), our three mechanisms - PIF, ReCIF and ReCIF02 - provide the best performance regarding content delivery rate. However, we observe that PIF02 deliv-



(a) Content delivery rate.



(b) Average delay.

Figure 5.5: Content packets delivered and average delivery delay for 30 consumers requesting popular content at different rates.

ers more contents with lower delay (Figure 5.7(b)) and with lower overhead (Figure 5.8). Thus, for the scenario with two consumers, it is worth to note that the PIF02 mechanism has the best overall performance.

As the number of consumers increases, the better performance of PIF02 becomes more noticeable. This performance can be proven by analyzing beyond the delivery rate, the average delay, and the overhead. This performance indicates that the restrictive

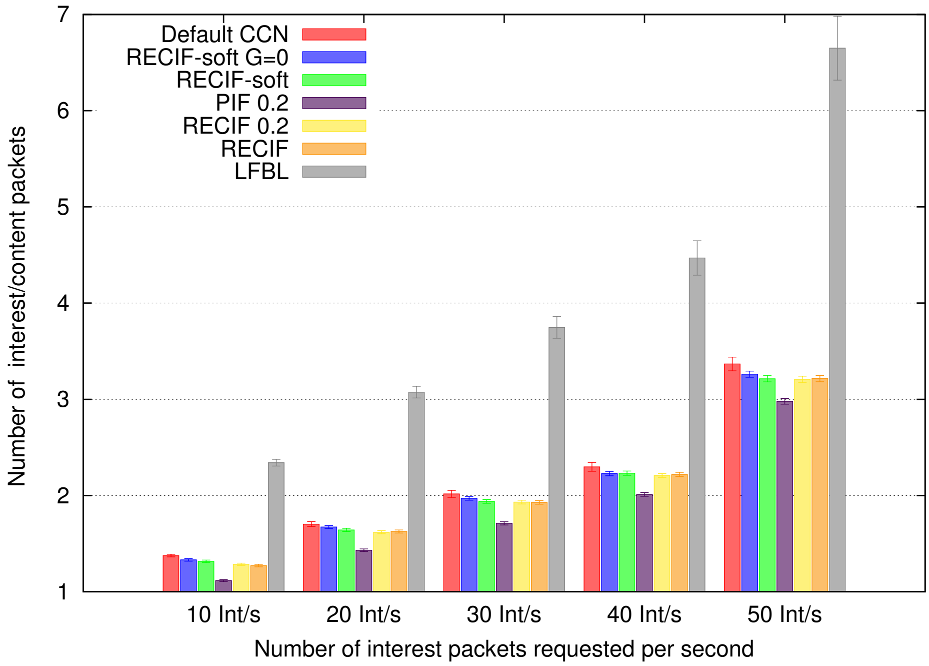


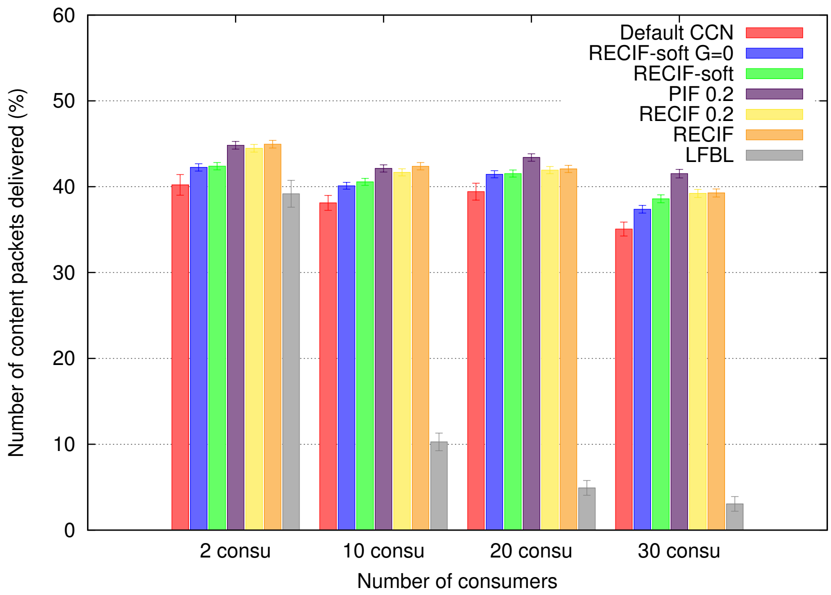
Figure 5.6: Overhead for 30 consumers requesting popular content at different rate.

forwarding criterion of PIF02 tends to be comparatively more efficient as the number of consumer increases in a scenario of sequential content requests.

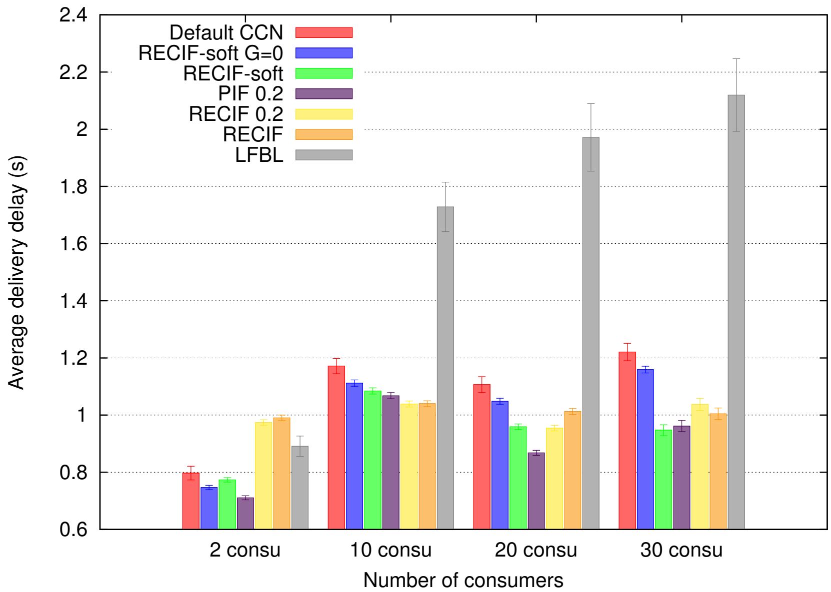
The performance of the LFBL is degrading as more consumers are added to the network. This degradation is a result of the number of flows per consumer. This multiple flows per consumer are because after the request flooding phase the further request would be directed to the producer ID. Thus, creating multiples flows between consumers and producers. Hence, LFBL presents its worst performance in delivery rate and average delay for 30 consumers.

We also vary the number of consumers considering these consumers are requesting contents by popularity, Figures 5.9 and 5.10. This scenario presents similarities, with the sequential scenario, regarding content delivery rate difference between default CCN and our mechanisms. However, we observe that the ReCIF and ReCIF02 mechanisms provide higher delivery rate as the number of consumer nodes increases. This fact, discussed before, is related to the less restrictive criterion of these mechanisms and therefore enables them to take advantage of in-network caching by retrieving the most popular contents from their near neighbors.

For two consumers requesting contents by popularity, we observe LFBL presents a similar performance to our mechanisms in relation to the content delivery rate. The reason for this is because with two consumers there would be only two flows from consumer to



(a) Content delivery rate.



(b) Average delay.

Figure 5.7: Content packets delivered and average delivery delay for 2, 10, 20 and 30 consumers requesting sequential content at 50 interest/s.

producers. The average delay is close to our mechanisms, in most cases. However, as in previous simulations, the increasing number of consumers degrades LFBL performance.

5.3.3 The Impact of Cache Size

We argue based on previous results the best performance of CCN against LFBL is due to the more efficient use of in-network caching. To verify this hypothesis we conduct

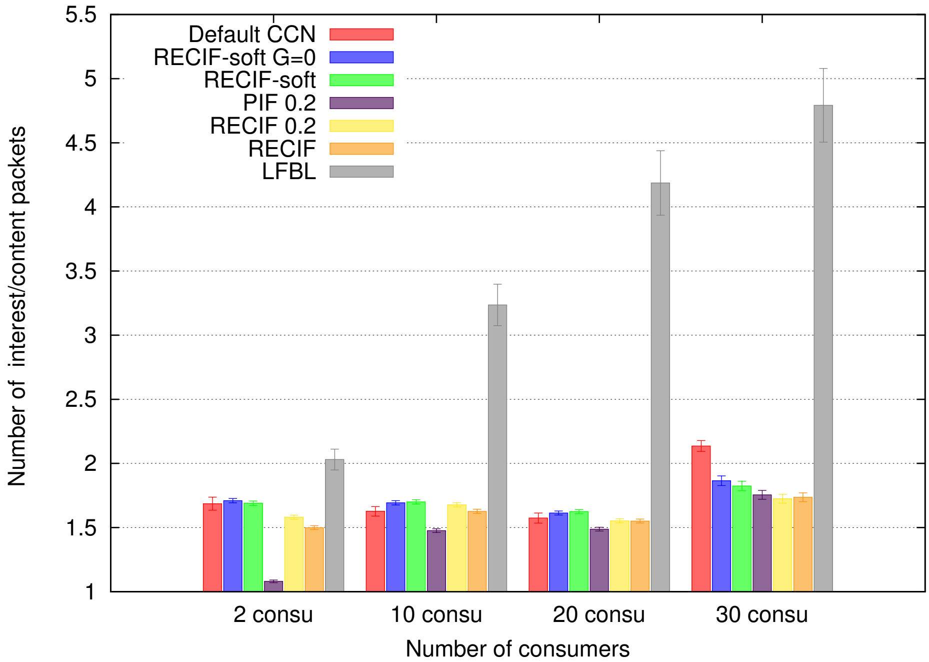
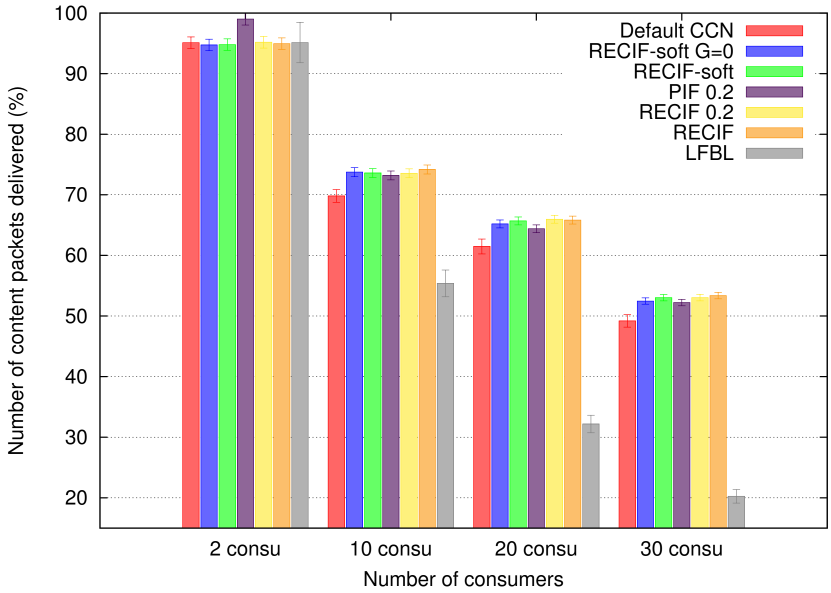


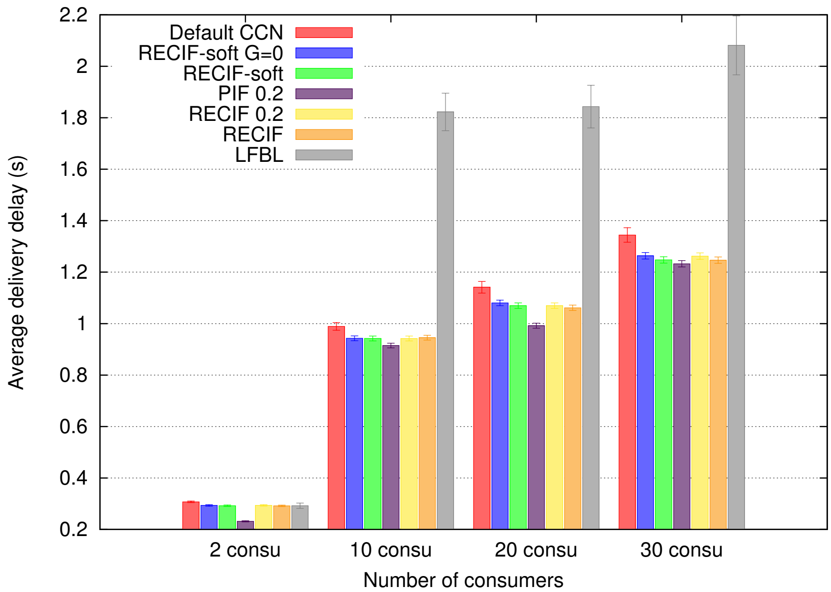
Figure 5.8: Overhead for 2, 10, 20 and 30 consumers requesting sequential content at 50 interest/s.

experiments considering a scenario with two consumers requesting contents sequentially and we vary the cache size of consumers, Figures 5.11 and 5.12. The reason to consider two consumers lies in the fact that for LFBL the greater the number of consumers the greater the competition for different flows. Therefore, with two consumers, we try to minimize interflow interference. We also consider consumers requesting contents sequentially because this configuration causes higher network saturation. With high saturation level, we have more packets to be forwarded, and thus, the forwarding mechanisms are triggered more often. Thus, we better observe the performance of these mechanisms.We vary the cache size of nodes from 10,000 to 50 contents. The first value is considered in previous simulations. To reduce that cache size, we observe the mean number of PIT entries during previous experiments and thus set the cache size to 200 contents. The idea is to choose a starting value for cache size limit, and thus decrease this size by half. The other cache size values are 100 and 50 content.

Figures 5.11 and 5.12 show that LFBL outperforms the default CCN and our protocols for minimal cache size and few consumers. We also observe that PIF02 mechanism reduces its performance compared to other mechanisms as the cache size decreases. However, it is worth mentioning that this extreme cache limitation was only induced to prove our hypothesis. For example, in our configuration, each content has 1 KB and thus a cache that stores 200 contents only requires 200 KB of memory space. This cache is not a hard



(a) Content delivery rate.



(b) Average delay.

Figure 5.9: Content packets delivered and average delivery delay for 2, 10, 20 and 30 consumers requesting popular content at 50 interest/s.

requirement for current network devices.

We conduct the same previous experiment, but now consumers request contents based on its popularity. Figures 5.13 and 5.14 show the results. First, we observe the impact of cache size reduction is smoother than in the previous experiments. In addition, PIF02 mechanism delivers more contents than the other mechanisms until the cache size reaches 100 contents because more contents are retrieved from caches of near nodes. If the cache

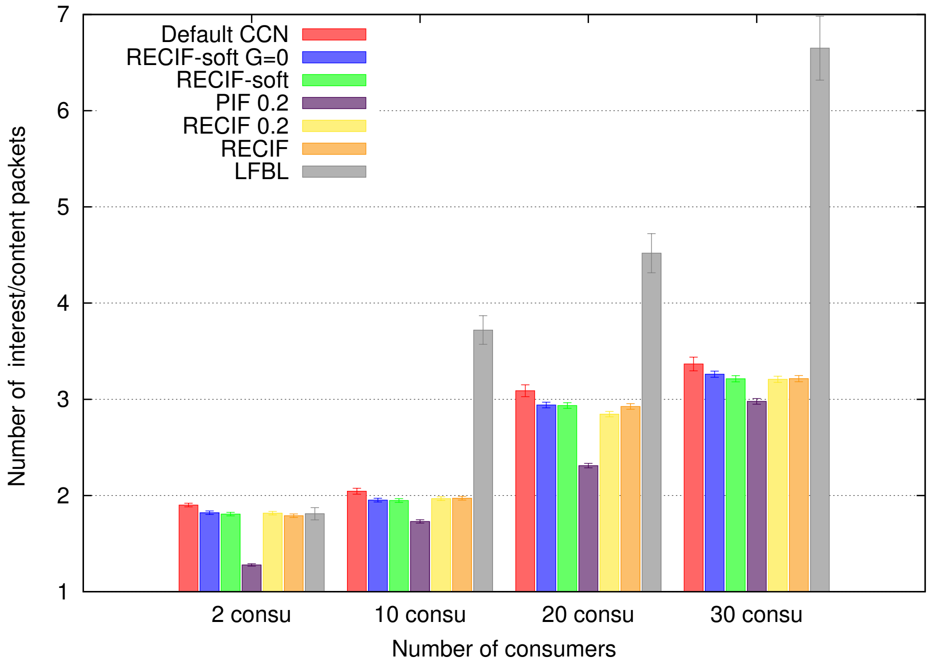
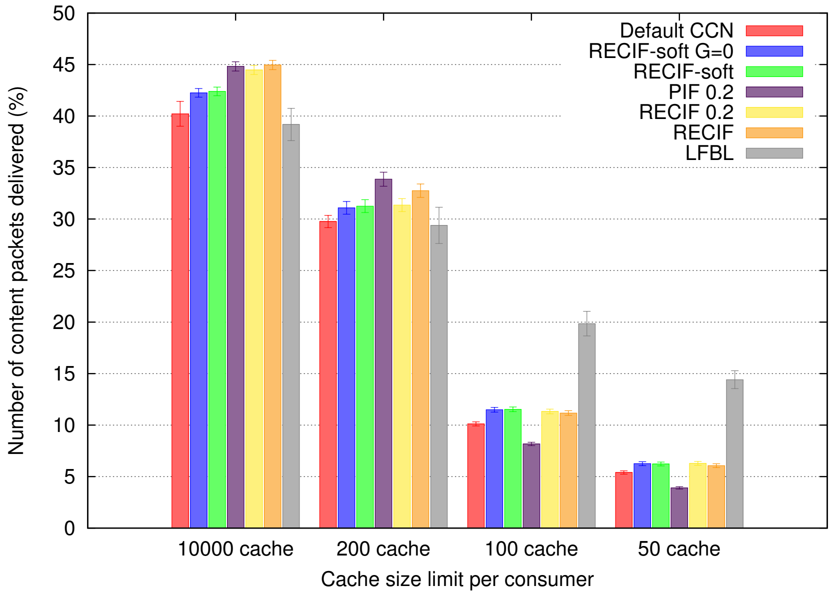


Figure 5.10: Overhead for 2, 10, 20 and 30 consumers requesting popular content at 50 interest/s.

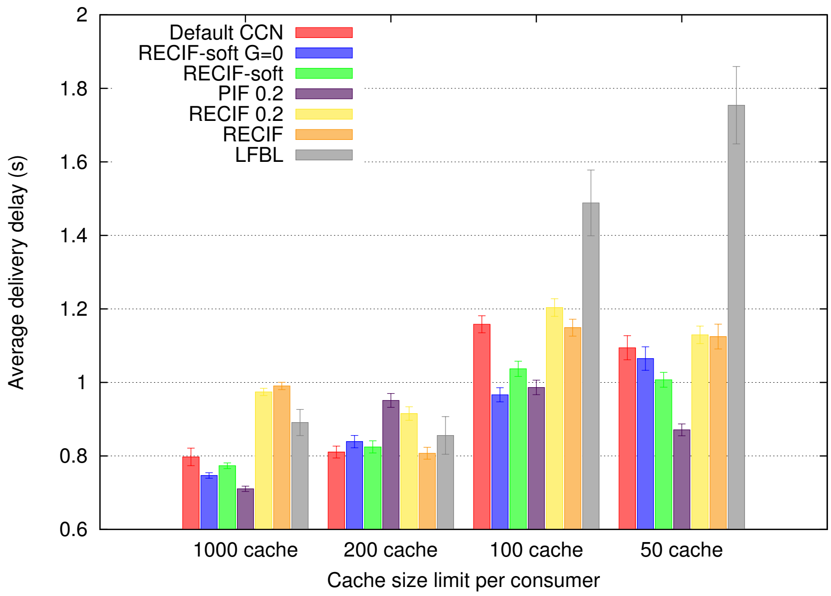
size is less than 100 contents, PIF02 starts to deliver fewer contents than all the other mechanisms. The reason for that relies on the combination of PIF02 forwarding criterion and fewer cache hits from near neighbors. Therefore, fewer neighbor nodes have popular contents in its small cache size. Hence, that restriction promotes less average delivery delay for PIF02 with a drawback of less content delivery.

LFBL performs better if the contents are requested sequentially than based on popularity. The reason for that lies in the fact that the other mechanisms can take advantage of popular contents stored in near nodes. Therefore, the strategy of limiting candidate nodes to deliver content is not effective for popular content until cache size is reduced to 50 contents. For this cache size, CCN strategies using content cache are less effective than LFBL.

Simulation results of our hypothesis lead us to conclude that when cache size becomes a high limiting factor, LFBL will present an improvement on the content delivery rate when compared to default CCN and our mechanisms. Therefore, regardless if consumers nodes are sequentially requesting contents or based on popularity, the cache size matters for our CCN mechanisms performance.



(a) Content delivery rate.



(b) Average delay.

Figure 5.11: Content packets delivered and average delivery delay for 2 consumers requesting sequential content with different cache sizes.

5.3.4 The Impact of the Number of Network Nodes

We conduct tests on Portland scenario with fewer network nodes. Therefore, we remove nodes located on the right side of the river from this scenario. The idea is to check the performance of our mechanisms in a less populated scenario. For this scenario, we use eight consumers and two producers nodes. We also increase the number of interest request from 10 to 50 interest per second. For this scenario, we conduct tests where the consumer

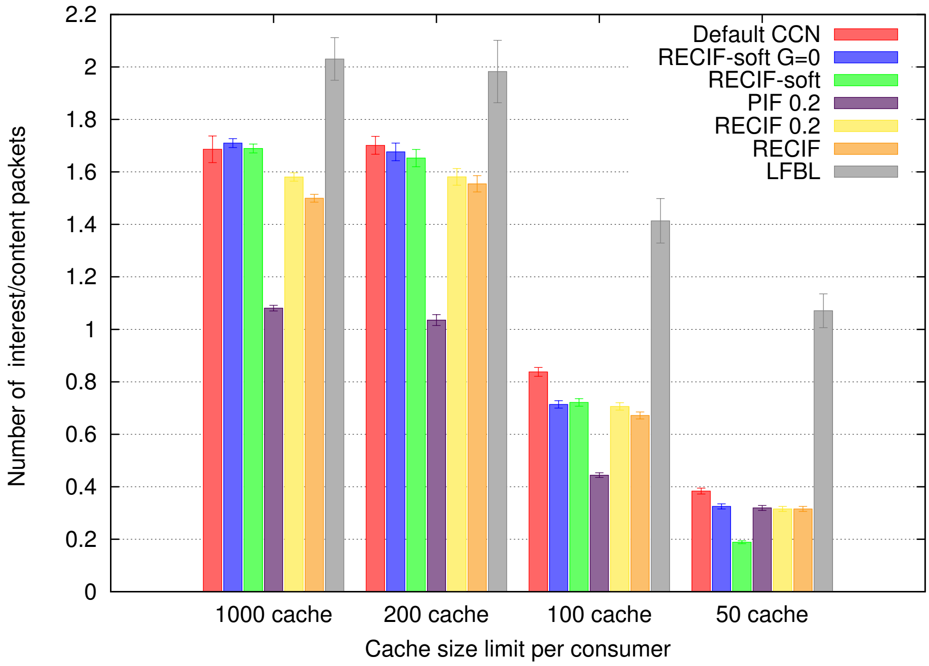


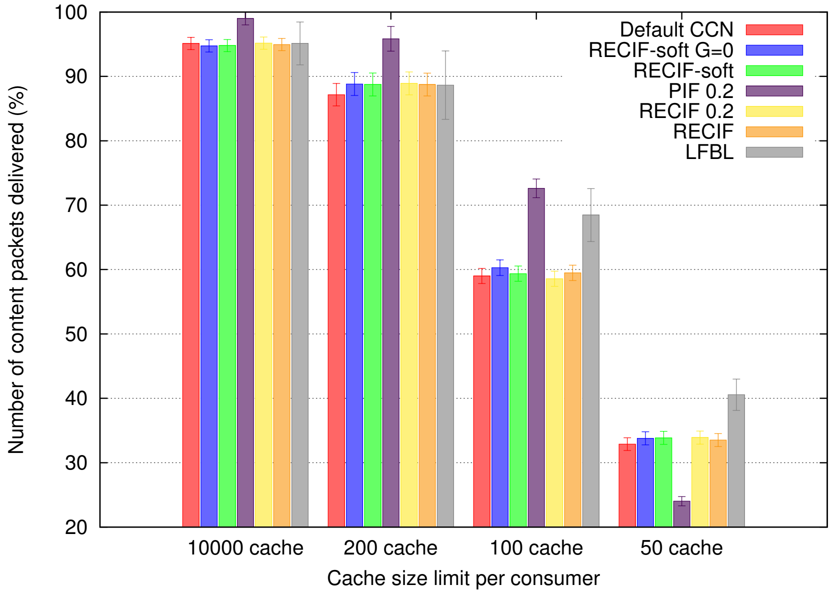
Figure 5.12: Overhead for 2 consumers requesting sequential content with different cache sizes.

requests content sequentially and with consumers requesting popular contents.

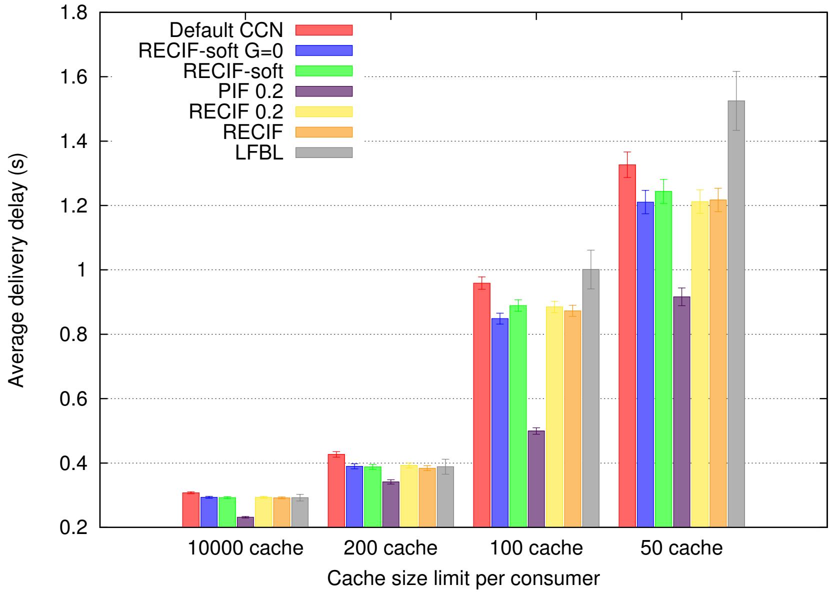
Concerning the number of content packets delivered, it is important to mention that RECIF-Soft presents the best performance when compared to the other mechanisms Figure 5.15(a). This performance gain occurs because this mechanism reduces the number of interest packets without being too restrictive as PIF02. Hence, RECIF-Soft achieves a better adaptative retransmit threshold to restrict the number of interest packets broadcast on the network. As the number of interest packet per second rises, RECIF-Soft delivers more content packets when compared to the other mechanisms. The gain regarding content packet delivered is 17,7% higher than default CCN and 393% higher than LFBL.

Regarding average delivery delay,Figure 5.15(b), RECIF-Soft presents a high performance than default CCN but when compared to the other mechanisms, for example, PIF02, it delivers with higher delay. The reason for this lies in the fact that more restrictive mechanism as PIF02 request less interest packet because more interest is filtered. Figure 5.16 presents this filter behavior where more restrictive mechanisms, i.e., PIF02 presents fewer interest/content packets.

For popular requests, RECIF-Soft also presents a higher performance when compared the other mechanisms, Figure 5.17 and 5.18. The reason for this is the same as in sequential content request scenario. Therefore RECIF-Soft delivers more contents by using an



(a) Content delivery rate.



(b) Average delay.

Figure 5.13: Content packets delivered and average delivery delay for 2 consumers requesting popular content with different cache sizes.

adaptively content filter. This filter behavior can be seen in Figure 5.18, where RECIF-Soft presents less overhead than default CCN but has a higher overhead then PIF02. It was expected from other results analyses that PIF02 presents more interest restriction, although in this less saturated scenario this restrictive approach is not ideal. The reason for this is because on a less saturated scenario a higher restrictive approach reduces the number of interest packet that reaches the producer nodes. Therefore this restrictive approach reduces the content delivery.

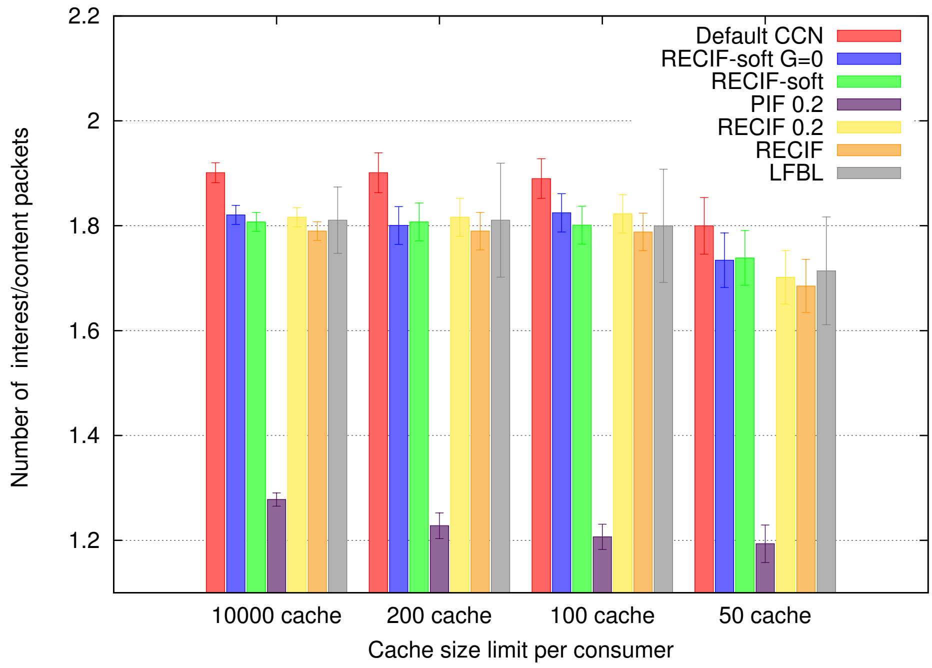
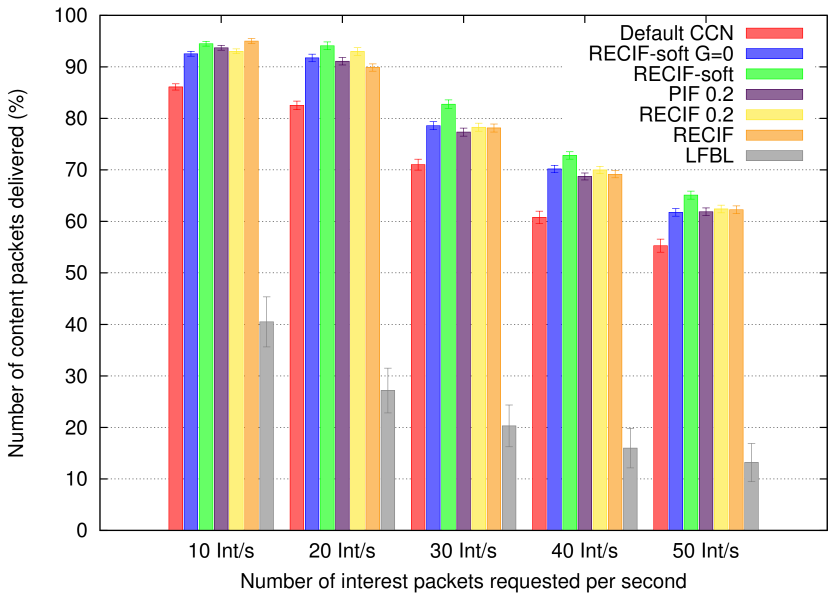


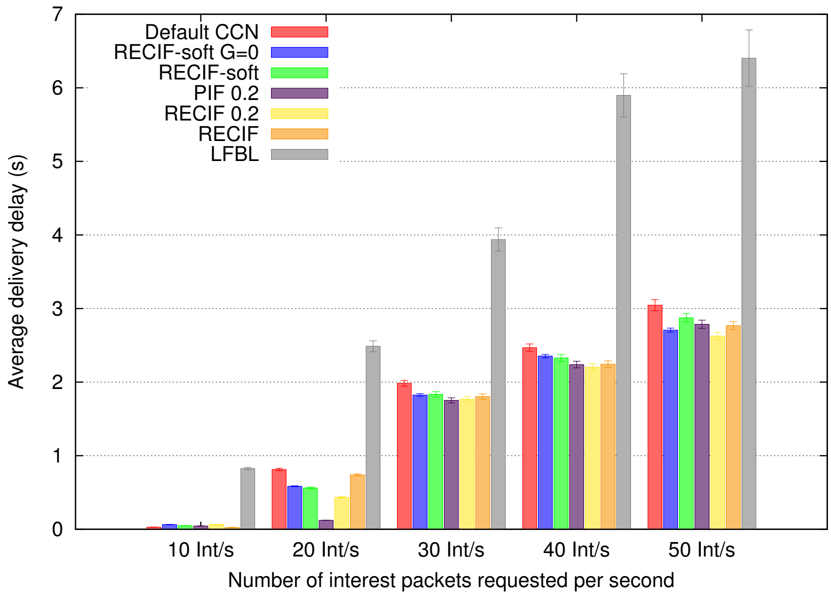
Figure 5.14: Overhead for 2 consumers requesting popular content with different cache sizes.

Rising the number of interest ratio, RECIF-Soft comparatively delivers more contents then default CCN. Thus, at the high ratio of 50 interest per second RECIF-Soft delivers 21,9% and 150,6% more contents than default CCN and LFBL respectively. RECIF-Soft also presents performance gain regarding content delivery delay when compared to default CCN and LFBL.

The – PIF, ReCIF, and ReCIF+PIF – mechanisms provide (a) higher content delivery as the network saturation level increases, (ii) less broadcast of interest packets to retrieve contents, (iii) lower average delivery delay. Nevertheless, some mechanisms are more suitable for less saturated environments, such as ReCIF, and others like PIF02 performs better in a saturated environment. The main reason for that behavior relies on the restriction approach used by the mechanisms. As a result, more restrictive criteria tend to be suitable for saturated networks. Conversely, for less saturated scenario RECIF-Soft mechanism presents a higher performance because its restriction criterion adapts more to the scenario. This flexible filter criterion, present on RECIF-Soft, provides higher content delivery without the less overhead performance presented on PIF02. Thus, less restricted and flexible criteria are suitable for less saturated networks.



(a) Content delivery rate.



(b) Average delay.

Figure 5.15: Content packets delivered and average delivery delay for 8 consumers requesting sequential content at different rates.

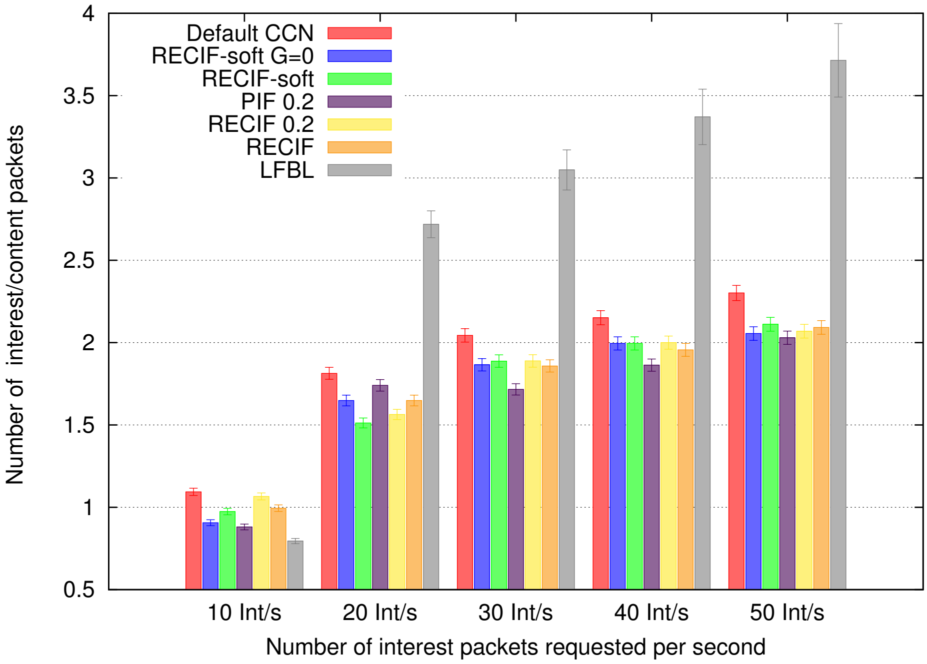
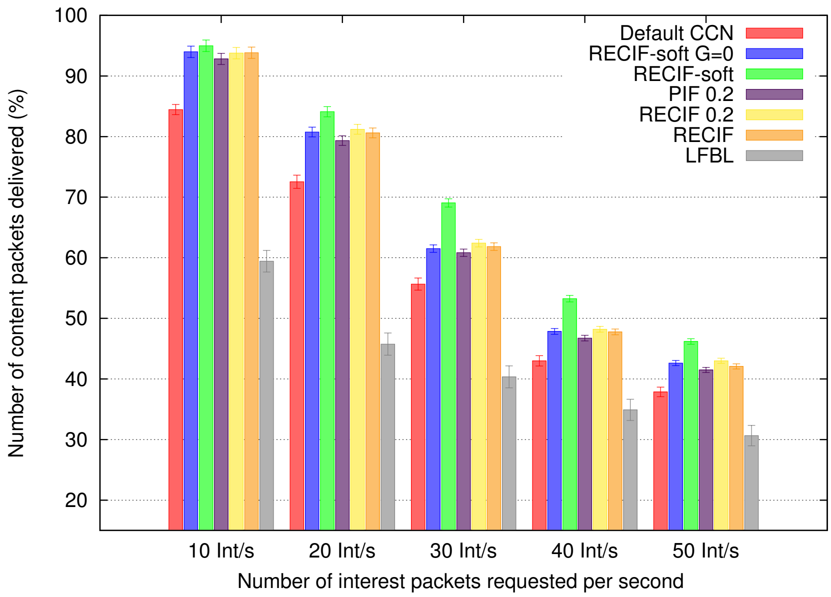
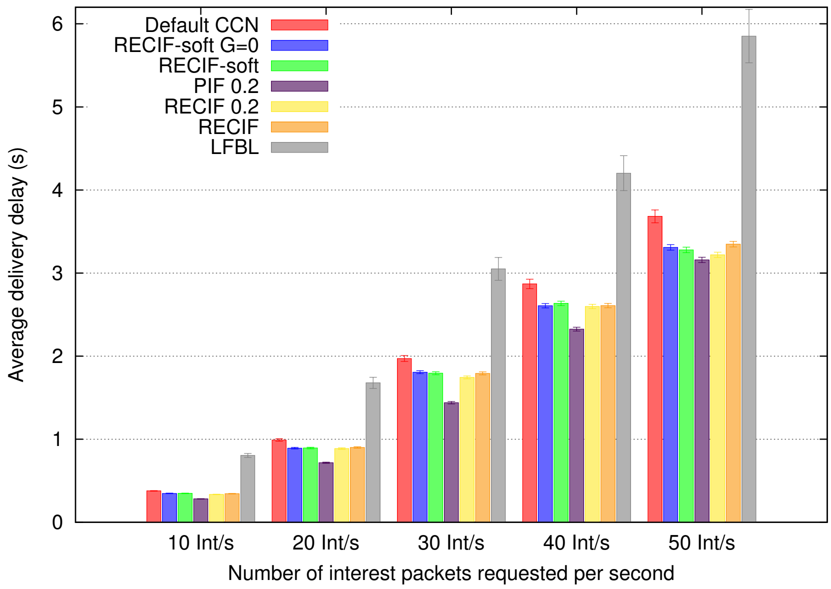


Figure 5.16: Overhead for 8 consumers requesting sequential content at different rates.



(a) Content delivery rate.



(b) Average delay.

Figure 5.17: Content packets delivered and average delivery delay for 8 consumers requesting popular content at different rates.

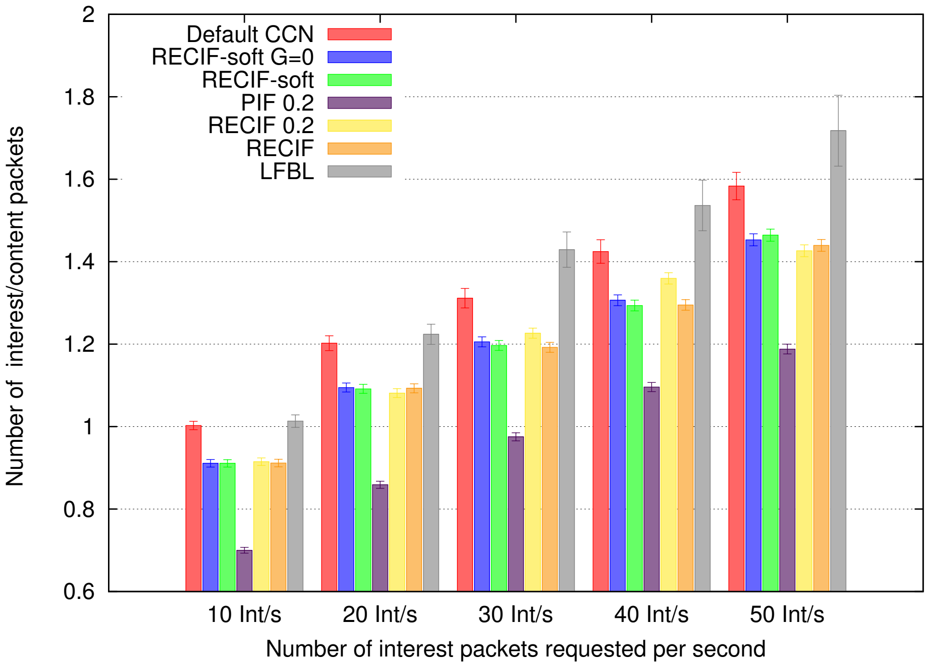


Figure 5.18: Overhead for 8 consumers requesting popular content at different rate.

Chapter 6

Conclusion

In this work, we present the challenges and issues concerning information-centric wireless mesh networks. We mainly focus on the broadcast storm problem, which is caused by the forwarding of interest packets. This problem starts with when each node broadcasts an interest packet received for the first time. Thereafter, every neighbor will repeat the same behavior leading into a broadcast storm of interest packets. The main reason for that behavior lies on the fact that all nodes have only one network interface. Thus, control strategies based on interfaces to reduce interest storm used in wired CCN are not able to work properly on CCN-based WMNs.

Although there are works that use interest restriction strategies. Those strategies present issues that may lead to a reduction in the overall network performance. This performance problem arises when the strategy to reduce interest packets does not consider multiple consumers and multiple producers for the same content. We conduct comparative experiments with our mechanisms and LFBL. We conduct experiments changing the content request rate and changing the number of consumers on the network to evaluate our mechanisms in saturated scenarios. Thus, increasing the number of consumer nodes for the same content allowed us to analyze the mechanism performance with multiple clients requests.

6.1 Contributions

We propose a set of mechanisms that limit the forwarding of interest packets to reduce the adverse effects of the broadcast storm in WMNs. The goal of the proposed mechanisms is to decrease the number of interest packets forwarded and thus decrease the probability of collisions caused by these packets. The first proposed mechanism is called

Probabilistic Interest Forwarding (PIF). With PIF, high-centrality nodes forward packets with probability p. Therefore, PIF reduces interest packet propagation on central nodes by reducing the number of packets that a central node may forwards. With the second one, called Retransmission-Counter-based Interest Forwarding (ReCIF), nodes use the number of hops traversed by an interest packet to forward or not this packet. ReCIF also has two operation modes, hard and soft. Both are used to define the retransmission threshold. This threshold is used to restrict the maximum number of forwarding hops that a packet may travel towards a producer node. The third proposed mechanism combines both criteria of PIF and ReCIF and is referred to as ReCIF + PIF. The idea is to guarantee that interest packets forwarded fewer times have a priority to be forwarded by a node and other interest packets are forwarded with probability p.

Our mechanisms are compared with the default CCN forwarding mechanism, Listen First Broadcast Later (LFBL) protocol and also with a TCP/IP-based network running the Optimized Link State Routing (OLSR) protocol. The evaluation results indicate that our mechanisms reduce the amount of interest packet diffused on the network. These mechanisms were implemented with a high level of interest packet restriction. Three of our mechanisms are focused on conduct restriction on nodes with high centrality degree and one mechanism variation was used to conduct interest restriction to all nodes. The reason for applying higher restrictions on central nodes derives from the probability of these nodes sending more interest packets, contributing more to broadcast storm. Results show the wireless CCN outperform Wireless Mesh Networks based on TCP/IP+OLSR, for the evaluated scenario, regarding content delivery rate and delay.

Results show that our proposed mechanisms are more efficient in scenarios with a higher number of hops between source and destination. For these scenarios, our mechanisms outperform default CCN mechanism by up to 21% regarding data delivery rate in dense scenarios with a high number of hops between source and destination and provides 25% lower delivery delay than the default CCN. One of our mechanisms, PIF02, outperforms LFBL in 1062% regarding data delivery rate with 55% less delivery delay on high saturation levels. On less saturated scenario, with a reduced part of Portland network, RECIF-Soft presents better performance regarding content delivery. This performance occurs because it adapts to the network less populated topology and filters interest packets without a highly restrict criterion regarding interest forwarding. Therefore, our mechanisms outperformance default CCN and LFBL in different levels of network saturation. Our mechanisms preset this performance regarding content delivery and delivery delay with less overhead on the network.

6.2 Future Works

For future works, we intend to test our mechanisms in other scenarios, including scenarios that present users mobility. The idea is to test our mechanisms on scenarios with pedestrian mobile users requesting and providing content using a WMNs. We will also conduct tests on VANETs scenarios and compare our mechanisms with others VANETs protocols.

Also, we will test modifications on our mechanisms to use centrality degree and betweenness. Furthermore, we will test the impact of filtering content packets and the content allocation on the network.

References

[1] Sandvine top 10 peak period applications. `https://www.sandvine.com/pr/2016/6/22/sandvine-report-netflix-encoding-optimizations-result-in-north-american-t` `html`. Accessed: 2017-07-15.

[2] Abolhasan, M.; Hagelstein, B.; Wang, J.-P. Real-world performance of current proactive multi-hop mesh protocols. In *Communications, 2009. APCC 2009. 15th Asia-Pacific Conference on* (2009), IEEE, pp. 44–47.

[3] Ade, S.; Tijare, P. Performance comparison of aodv, dsdv, olsr and dsr routing protocols in mobile ad hoc networks. *International journal of information technology and knowledge management 2*, 2 (2010), 545–548.

[4] Adnane, A.; Bidan, C.; de Sousa Júnior, R. T. Trust-based security for the olsr routing protocol. *Computer Communications 36*, 10 (2013), 1159–1171.

[5] Afanasyev, A.; Moiseenko, I.; Zhang, L., et al. ndnsim: Ndn simulator for ns-3. *Named Data Networking (NDN) Project, Tech. Rep. NDN-0005, Rev 2* (2012).

[6] Ahlgren, B.; D'Ambrosio, M.; Marchisio, M.; Marsh, I.; Dannewitz, C.; Ohlman, B.; Pentikousis, K.; Strandberg, O.; Rembarz, R.; Vercellone, V. Design considerations for a network of information. In *Proceedings of the 2008 ACM CoNEXT Conference* (2008), ACM, p. 66.

[7] Ahlgren, B.; Dannewitz, C.; Imbrenda, C.; Kutscher, D.; Ohlman, B. A survey of information-centric networking. *IEEE Communications Magazine 50*, 7 (2012).

[8] Akhshabi, S.; Begen, A. C.; Dovrolis, C. An experimental evaluation of rate-adaptation algorithms in adaptive streaming over http. In *Proceedings of the second annual ACM conference on Multimedia systems* (2011), ACM, pp. 157–168.

[9] Akkaya, K.; Younis, M. A survey on routing protocols for wireless sensor networks. *Ad hoc networks 3*, 3 (2005), 325–349.

[10] Akyildiz, I. F.; Wang, X. A survey on wireless mesh networks. *IEEE Communications magazine 43*, 9 (2005), S23–S30.

[11] Akyildiz, I. F.; Wang, X.; Wang, W. Wireless mesh networks: a survey. *Computer networks 47*, 4 (2005), 445–487.

[12] Al-Arnaout, Z.; Fu, Q.; Frean, M. A content replication scheme for wireless mesh networks. In *Proceedings of the 22nd international workshop on Network and Operating System Support for Digital Audio and Video* (2012), ACM, pp. 39–44.

[13] Amadeo, M.; Campolo, C.; Molinaro, A. Content-centric networking: is that a solution for upcoming vehicular networks? In *Proceedings of the ninth ACM international workshop on Vehicular inter-networking, systems, and applications* (2012), ACM, pp. 99–102.

[14] Amadeo, M.; Campolo, C.; Molinaro, A. Multi-source data retrieval in iot via named data networking. In *Proceedings of the 1st international conference on Information-centric networking* (2014), ACM, pp. 67–76.

[15] Amadeo, M.; Campolo, C.; Molinaro, A. Named data networking for priority-based content dissemination in vanets. In *Personal, Indoor, and Mobile Radio Communications (PIMRC), 2016 IEEE 27th Annual International Symposium on* (2016), IEEE, pp. 1–6.

[16] Amadeo, M.; Molinaro, A. Chanet: A content-centric architecture for ieee 802.11 manets. In *Network of the Future (NOF), 2011 International Conference on the* (2011), IEEE, pp. 122–127.

[17] Amadeo, M.; Molinaro, A.; Campolo, C.; Sifalakis, M.; Tschudin, C. Transport layer design for named data wireless networking. In *Computer Communications Workshops (INFOCOM WKSHPS), 2014 IEEE Conference on* (2014), IEEE, pp. 464–469.

[18] Badov, M.; Seetharam, A.; Kurose, J.; Firoiu, V.; Nanda, S. Congestion-aware caching and search in information-centric networks. In *Proceedings of the 1st international conference on Information-centric networking* (2014), ACM, pp. 37–46.

[19] Bastug, E.; Bennis, M.; Debbah, M. Living on the edge: The role of proactive caching in 5g wireless networks. *IEEE Communications Magazine 52*, 8 (2014), 82–89.

[20] Benzaid, M.; Minet, P.; Al Agha, K. Integrating fast mobility in the olsr routing protocol. In *Mobile and Wireless Communications Network, 2002. 4th International Workshop on* (2002), IEEE, pp. 217–221.

[21] Bicket, J.; Aguayo, D.; Biswas, S.; Morris, R. Architecture and evaluation of an unplanned 802.11 b mesh network. In *Proceedings of the 11th annual international conference on Mobile computing and networking* (2005), ACM, pp. 31–42.

[22] Bosunia, M. R.; Kim, A.; Jeong, D. P.; Park, C.; Jeong, S.-H. Enhanced multimedia data delivery based on content-centric networking in wireless networks. *Appl. Math 9*, 2L (2015), 579–589.

[23] Bruno, R.; Conti, M.; Gregori, E. Mesh networks: commodity multihop ad hoc networks. *IEEE communications magazine 43*, 3 (2005), 123–131.

[24] Burresi, S.; Canali, C.; Renda, M. E.; Santi, P. Meshchord: a location-aware, cross-layer specialization of chord for wireless mesh networks (concise contribution). In *Pervasive Computing and Communications, 2008. PerCom 2008. Sixth Annual IEEE International Conference on* (2008), IEEE, pp. 206–212.

[25] Caesar, M.; Condie, T.; Kannan, J.; Lakshminarayanan, K.; Stoica, I.; Shenker, S. Rofl: routing on flat labels. *ACM SIGCOMM Computer Communication Review 36*, 4 (2006), 363–374.

[26] Campista, M. E. M.; Esposito, P. M.; Moraes, I. M.; Costa, L. H. M. K.; Duarte, O. C. M. B.; Passos, D. G.; de Albuquerque, C. N.; Saade, D. C. M.; Rubinstein, M. G. Routing metrics and protocols for wireless mesh networks. *Network, IEEE 22*, 1 (2008), 6–12.

[27] Canali, C.; Renda, M. E.; Santi, P.; Burresi, S. Enabling efficient peer-to-peer resource sharing in wireless mesh networks. *Mobile Computing, IEEE Transactions on 9*, 3 (2010), 333–347.

[28] Carofiglio, G.; Gallo, M.; Muscariello, L. Icp: Design and evaluation of an interest control protocol for content-centric networking. In *Computer Communications Workshops (INFOCOM WKSHPS), 2012 IEEE Conference on* (2012), IEEE, pp. 304–309.

[29] Chatzimisios, P.; Boucouvalas, A.; Vitsas, V. Effectiveness of rts/cts handshake in ieee 802.11 a wireless lans. *Electronics Letters 40*, 14 (2004), 915–916.

[30] Chatzimisios, P.; Boucouvalas, A.; Vitsas, V. Optimisation of rts/cts handshake in ieee 802.11 wireless lans for maximum performance. In *Global Telecommunications Conference Workshops, 2004. GlobeCom Workshops 2004. IEEE* (2004), IEEE, pp. 270–275.

[31] Clausen, T.; Jacquet, P. Optimized link state routing protocol (olsr). Tech. rep., 2003.

[32] Committee, I. C. S. L. M. S., et al. Wireless lan medium access control (mac) and physical layer (phy) specifications. *IEEE Standard 802.11-1997* (1997).

[33] Das, S. M.; Pucha, H.; Papagiannaki, K.; Hu, Y. C. Studying wireless routing link metric dynamics. In *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement* (2007), ACM, pp. 327–332.

[34] De Brito, G. M.; Velloso, P. B.; Moraes, I. M. *Information Centric Networks: A New Paradigm for the Internet.* John Wiley & Sons, 2013.

[35] de Brito, G. M.; Velloso, P. B.; Moraes, I. M. Uma análise do desempenho de redes sem-fio orientadas a conteúdo. *Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos (SBRC)* (2014).

[36] De Couto, D. S.; Aguayo, D.; Bicket, J.; Morris, R. A high-throughput path metric for multi-hop wireless routing. *Wireless Networks 11*, 4 (2005), 419–434.

[37] De Rango, F.; Fotino, M.; Marano, S. Ee-olsr: energy efficient olsr routing protocol for mobile ad-hoc networks. In *Military Communications Conference, 2008. MILCOM 2008. IEEE* (2008), IEEE, pp. 1–7.

[38] Detti, A.; Blefari Melazzi, N.; Salsano, S.; Pomposini, M. Conet: a content centric inter-networking architecture. In *Proceedings of the ACM SIGCOMM workshop on Information-centric networking* (2011), ACM, pp. 50–55.

[39] Detti, A.; Tassetto, D.; Melazzi, N. B.; Fedi, F. Exploiting content centric networking to develop topic-based, publish–subscribe manet systems. *Ad Hoc Networks 24* (2015), 115–133.

[40] Draves, R.; Padhye, J.; Zill, B. Routing in multi-radio, multi-hop wireless mesh networks. In *Proceedings of the 10th annual international conference on Mobile computing and networking* (2004), ACM, pp. 114–128.

[41] Esposito, P. M.; Campista, M. E. M.; Moraes, I. M.; Costa, L. H. M.; Duarte, O. C. M.; Rubinstein, M. G. Implementing the expected transmission time metric for olsr wireless mesh networks. In *Wireless Days, 2008. WD'08. 1st IFIP* (2008), IEEE, pp. 1–5.

[42] Fotiou, N.; Nikander, P.; Trossen, D.; Polyzos, G. C., et al. Developing information networking further: From psirp to pursuit. In *Broadnets* (2010), Springer, pp. 1–13.

[43] Garcia-Reinoso, J.; Fernández, N.; Vidal, I.; Fisteus, J. A. Scalable data replication in content-centric networking based on alias names. *Journal of Network and Computer Applications 47* (2015), 85–98.

[44] Ge, Y.; Kunz, T.; Lamont, L. Quality of service routing in ad-hoc networks using olsr. In *System Sciences, 2003. Proceedings of the 36th Annual Hawaii International Conference on* (2003), IEEE, pp. 9–pp.

[45] Golrezaei, N.; Molisch, A. F.; Dimakis, A. G.; Caire, G. Femtocaching and device-to-device collaboration: A new architecture for wireless video distribution. *IEEE Communications Magazine 51*, 4 (2013), 142–149.

[46] Gritter, M.; Cheriton, D. R. An architecture for content routing support in the internet. In *USITS* (2001), vol. 1, pp. 4–4.

[47] Han, H.; Wu, M.; Hu, Q.; Wang, N. Best route, error broadcast: A content-centric forwarding protocol for manets. In *Vehicular Technology Conference (VTC Fall), 2014 IEEE 80th* (2014), IEEE, pp. 1–5.

[48] Jacobson, V.; Smetters, D. K.; Thornton, J. D.; Plass, M. F.; Briggs, N. H.; Braynard, R. L. Networking named content. In *Proceedings of the 5th international conference on Emerging networking experiments and technologies* (2009), ACM, pp. 1–12.

[49] Ji, M.; Caire, G.; Molisch, A. F. Wireless device-to-device caching networks: Basic principles and system performance. *IEEE Journal on Selected Areas in Communications 34*, 1 (2016), 176–189.

[50] Jmal, R.; Simon, G.; Chaari, L. Network-assisted strategy for dash over ccn.

[51] Johnson, D.; Hancke, G. Comparison of two routing metrics in olsr on a grid based mesh network. *Ad Hoc Networks 7*, 2 (2009), 374–387.

[52] Johnson, D. B.; Maltz, D. A.; Broch, J., et al. Dsr: The dynamic source routing protocol for multi-hop wireless ad hoc networks. *Ad hoc networking 5* (2001), 139–172.

[53] Juluri, P.; Tamarapalli, V.; Medhi, D. Qoe management in dash systems using the segment aware rate adaptation algorithm. In *Network Operations and Management Symposium (NOMS), 2016 IEEE/IFIP* (2016), IEEE, pp. 129–136.

[54] Kim, D.; Kim, J.-h.; Moon, C.; Choi, J.; Yeom, I. Efficient content delivery in mobile ad-hoc networks using ccn. *Ad Hoc Networks 36* (2016), 81–99.

[55] Koponen, T.; Chawla, M.; Chun, B.-G.; Ermolinskiy, A.; Kim, K. H.; Shenker, S.; Stoica, I. A data-oriented (and beyond) network architecture. In *ACM SIGCOMM Computer Communication Review* (2007), vol. 37, ACM, pp. 181–192.

[56] Kowalik, K.; Davis, M. Why are there so many routing protocols for wireless mesh networks. In *Irish Signal and Systems Conference* (2006), pp. 1–5.

[57] Lagutin, D.; Visala, K.; Tarkoma, S. Publish/subscribe for internet: Psirp perspective. *Future internet assembly 84* (2010).

[58] Lederer, S.; Mueller, C.; Rainer, B.; Timmerer, C.; Hellwagner, H. Adaptive streaming over content centric networks in mobile networks using multiple links. In *Communications Workshops (ICC), 2013 IEEE International Conference on* (2013), IEEE, pp. 677–681.

[59] Li, B.; Ma, M.; Yang, X. Advanced perceptive forwarding in content-centric networks. *IEEE Access 5* (2017), 4595–4605.

[60] Li, F. Y.; Vandoni, L.; Zicca, G.; Zanoli, S. Olsr mesh networks for broadband access: Enhancements, implementation and deployment. In *Circuits and Systems for Communications, 2008. ICCSC 2008. 4th IEEE International Conference on* (2008), IEEE, pp. 802–806.

[61] Li, Z.; Simon, G. Cooperative caching in a content centric network for video stream delivery. *Journal of network and systems management 23*, 3 (2015), 445–473.

[62] Ligo, A. K.; Peha, J. M.; Barros, J. Throughput and cost-effectiveness of vehicular mesh networks for internet access. In *Vehicular Technology Conference (VTC-Fall), 2016 IEEE 84th* (2016), IEEE, pp. 1–7.

[63] Liu, D.; Chen, B.; Yang, C.; Molisch, A. F. Caching at the wireless edge: design aspects, challenges, and future directions. *IEEE Communications Magazine 54*, 9 (2016), 22–28.

[64] Liu, Y.; Geurts, J.; Point, J.-C.; Lederer, S.; Rainer, B.; Muller, C.; Timmerer, C.; Hellwagner, H. Dynamic adaptive streaming over ccn: a caching and overhead analysis. In *Communications (ICC), 2013 IEEE International Conference on* (2013), IEEE, pp. 3629–3633.

[65] Liu, Y.; Zhu, D.; Ma, W. A novel cooperative caching scheme for content centric mobile ad hoc networks. In *Computers and Communication (ISCC), 2016 IEEE Symposium on* (2016), IEEE, pp. 824–829.

[66] Lu, Y.; Li, X.; Yu, Y.-T.; Gerla, M. Information-centric delay-tolerant mobile ad-hoc networks. In *Computer Communications Workshops (INFOCOM WKSHPS), 2014 IEEE Conference on* (2014), IEEE, pp. 428–433.

[67] Mangili, M.; Martignon, F.; Paraboschi, S. A cache-aware mechanism to enforce confidentiality, trackability and access policy evolution in content-centric networks. *Computer Networks 76* (2015), 126–145.

[68] Mascarenhas, D.; Rubinstein, M.; Sztajnberg, A. Uma nova métrica para protocolos de roteamento em redes em malha sem fio. *XXVI Simpósio Brasileiro de Redes de Computadores-SBrT* (2008).

[69] Mascarenhas, D. M.; Moraes, I. M. Mecanismos de difusão limitada de interesses em redes em malha sem-fio orientadas a conteúdo. *WPerformance XIII Workshop em Desempenho de Sistemas Computacionais e de Comunicacao* (2014).

[70] Matos Mascarenhas, D.; Monteiro Moraes, I. Limiting the interest-packet forwarding in information-centric wireless mesh networks. In *Wireless Days (WD), 2014 IFIP* (2014), IEEE, pp. 1–6.

[71] Meisel, M.; Pappas, V.; Zhang, L. Ad hoc networking via named data. In *Proceedings of the fifth ACM international workshop on Mobility in the evolving internet architecture* (2010), ACM, pp. 3–8.

[72] Meisel, M.; Pappas, V.; Zhang, L. Listen first, broadcast later: Topology-agnostic forwarding under high dynamics. In *Annual conference of international technology alliance in network and information science* (2010), p. 8.

[73] Mohandas, G.; Silas, S.; Sam, S. Survey on routing protocols on mobile adhoc networks. In *Automation, Computing, Communication, Control and Compressed Sensing (iMac4s), 2013 International Multi-Conference on* (2013), IEEE, pp. 514–517.

[74] Molisch, A. F.; Caire, G.; Ott, D.; Foerster, J. R.; Bethanabhotla, D.; Ji, M. Caching eliminates the wireless bottleneck in video aware wireless networks. *Advances in Electrical Engineering 2014* (2014).

[75] Nguyen, D.; Minet, P. Analysis of mpr selection in the olsr protocol. In *Advanced Information Networking and Applications Workshops, 2007, AINAW'07. 21st International Conference on* (2007), vol. 2, IEEE, pp. 887–892.

[76] Oh, S.-Y.; Lau, D.; Gerla, M. Content centric networking in tactical and emergency manets. In *Wireless Days (WD), 2010 IFIP* (2010), IEEE, pp. 1–5.

[77] Passos, D.; Teixeira, D. V.; Muchaluat-Saade, D. C.; Magalhães, L. C. S.; Albuquerque, C. Mesh network performance measurements. In *International Information and Telecommunicatios Technologies Symposium (I2TS)* (2006), pp. 48–55.

[78] Pathak, P. H.; Dutta, R. A survey of network design problems and joint design approaches in wireless mesh networks. *Communications Surveys & Tutorials, IEEE 13*, 3 (2011), 396–428.

[79] Perino, D.; Varvello, M. A reality check for content centric networking. In *Proceedings of the ACM SIGCOMM workshop on Information-centric networking* (2011), ACM, pp. 44–49.

[80] Perkins, C.; Belding-Royer, E.; Das, S. Ad hoc on-demand distance vector (aodv) routing. Tech. rep., 2003.

[81] Posch, D.; Kreuzberger, C.; Rainer, B.; Hellwagner, H. Using in-network adaptation to tackle inefficiencies caused by dash in information-centric networks. In *Proceedings of the 2014 Workshop on Design, Quality and Deployment of Adaptive Video Streaming* (2014), ACM, pp. 25–30.

[82] Poularakis, K.; Iosifidis, G.; Argyriou, A.; Koutsopoulos, I.; Tassiulas, L. Caching and operator cooperation policies for layered video content delivery. In *IEEE International Conference on Computer Communications (INFOCOM)* (2016).

[83] Psaras, I.; Chai, W. K.; Pavlou, G. Probabilistic in-network caching for information-centric networks. In *Proceedings of the second edition of the ICN workshop on Information-centric networking* (2012), ACM, pp. 55–60.

[84] Rainer, B.; Posch, D.; Hellwagner, H. Investigating the performance of pull-based dynamic adaptive streaming in ndn. *IEEE Journal on Selected Areas in Communications 34*, 8 (2016), 2130–2140.

[85] Rainer, B.; Posch, D.; Leibetseder, A.; Theuermann, S.; Hellwagner, H. A low-cost ndn testbed on banana pi routers. *IEEE Communications Magazine 54*, 9 (2016), 105–111.

[86] Rubinstein, M.; Moraes, I.; Campista, M.; Costa, L.; Duarte, O. A survey on wireless ad hoc networks. *Mobile and Wireless Communication Networks* (2006), 1–33.

[87] Saade, D.; Albuquerque, C.; Magalhaes, L.; Passos, D.; Duarte, J.; Valle, R. Redes em malha: Soluçao de baixo custo para popularizaçao do acesso a internet no brasil. *XXV Simpósio Brasileiro de Telecomunicações-SBrT* (2007), 1–6.

[88] Sbai, M. K.; Barakat, C.; Choi, J.; Al Hamra, A.; Turletti, T. Adapting bittorrent to wireless ad hoc networks. In *Ad-hoc, Mobile and Wireless Networks*. Springer, 2008, pp. 189–203.

[89] Schneider, K. M.; Mast, K.; Krieger, U. R. Ccn forwarding strategies for multihomed mobile terminals. In *Networked Systems (NetSys), 2015 International Conference and Workshops on* (2015), IEEE, pp. 1–5.

[90] Sengul, C.; Viana, A. C.; Ziviani, A. A survey of adaptive services to cope with dynamics in wireless self-organizing networks. *ACM Computing Surveys (CSUR) 44*, 4 (2012), 23.

[91] Senior, R.; Phillips, C. CRAWDAD data set pdx/metrofi (v. 2011-10-24). Downloaded from http://crawdad.org/pdx/metrofi/, Oct. 2011.

[92] Shi, T.; Zhang, L.; Xu, C. Research on the energy-efficient caching strategy in content-centric networking. In *2015 Asia-Pacific Energy Equipment Engineering Research Conference* (2015), Atlantis Press.

[93] Tatar, A.; de Amorim, M. D.; Fdida, S.; Antoniadis, P. A survey on predicting the popularity of web content. *Journal of Internet Services and Applications 5*, 1 (2014), 1.

[94] Thomas, E.; van Deventer, M.; Stockhammer, T.; Begen, A. C.; Famaey, J. Enhancing mpeg dash performance via server and network assistance.

[95] Tian, G.; Liu, Y. Towards agile and smooth video adaptation in http adaptive streaming. *IEEE/ACM Transactions on Networking 24*, 4 (2016), 2386–2399.

[96] Toutouh, J.; García-Nieto, J.; Alba, E. Intelligent olsr routing protocol optimization for vanets. *IEEE transactions on vehicular technology 61*, 4 (2012), 1884–1894.

[97] Tseng, Y.-C.; Ni, S.-Y.; Chen, Y.-S.; Sheu, J.-P. The broadcast storm problem in a mobile ad hoc network. *Wireless networks 8*, 2-3 (2002), 153–167.

[98] Wang, L.; Afanasyev, A.; Kuntz, R.; Vuyyuru, R.; Wakikawa, R.; Zhang, L. Rapid traffic information dissemination using named data. In *Proceedings of the 1st ACM workshop on Emerging Name-Oriented Mobile Networking Design-Architecture, Algorithms, and Applications* (2012), ACM, pp. 7–12.

[99] Wang, L.; Hoque, A.; Yi, C.; Alyyan, A.; Zhang, B. Ospfn: An ospf based routing protocol for named data networking. *University of Memphis and University of Arizona, Tech. Rep* (2012).

[100] Wei, H.-Y.; Kim, S.; Ganguly, S.; Izmailov, R. Seamless handoff support in wireless mesh networks. In *Operator-Assisted (Wireless Mesh) Community Networks, 2006 1st Workshop on* (2006), IEEE, pp. 1–8.

[101] Yamazaki, T.; Yamamoto, R.; Miyoshi, T.; Asaka, T.; Tanaka, Y. Opportunistic routing using prioritized forwarders with retransmission control. *Image Processing & Communications 21*, 1 (2016), 57–66.

[102] Yang, Y.; Wang, J.; Kravets, R. Designing routing metrics for mesh networks. In *IEEE Workshop on Wireless Mesh Networks (WiMesh)* (2005).

[103] Yang, Y.; Wang, J.; Kravets, R. Interference-aware load balancing for multihop wireless networks. Tech. rep., 2005.

[104] Yu, Y.-T.; Dilmaghani, R. B.; Calo, S.; Sanadidi, M.; Gerla, M. Interest propagation in named data manets. In *Computing, Networking and Communications (ICNC), 2013 International Conference on* (2013), IEEE, pp. 1118–1122.

[105] Yu, Y.-T.; Li, Y.; Ma, X.; Shang, W.; Sanadidi, M.; Gerla, M. Scalable opportunistic vanet content routing with encounter information. In *Network Protocols (ICNP), 2013 21st IEEE International Conference on* (2013), IEEE, pp. 1–6.

[106] Zeydan, E.; Baştuğ, E.; Bennis, M.; Kader, M. A.; Karatepe, A.; Er, A. S.; Debbah, M. Big data caching for networking: Moving from cloud to edge. *arXiv preprint arXiv:1606.01581* (2016).

[107] Zhang, C.; Wang, L.; Wang, Z.; Wang, S. A service-based congestion control strategy for content centric networking. In *2015 International Symposium on Computers & Informatics* (2015), Atlantis Press.

[108] Zhao, N.; Liu, X.; Yu, F. R.; Li, M.; Leung, V. C. Communications, caching, and computing oriented small cell networks with interference alignment. *IEEE Communications Magazine 54*, 9 (2016), 29–35.

[109] Zhu, J.; Guo, X.; Yang, L. L.; Conner, W. S. Leveraging spatial reuse in 802.11 mesh networks with enhanced physical carrier sensing. In *Communications, 2004 IEEE International Conference on* (2004), vol. 7, IEEE, pp. 4004–4011.

APPENDIX A – OLSR and ETX

The next sections briefly describe Optimized Link State Routing protocol and the ETX metric.

A.1 Optimized Link State Routing (OLSR)

The OLSR protocol [OLSR, 2008] has good performance in large and densely populated wireless networks, mainly because it employs Multipoint Relays (MPRs). The use of MPRs reduces the number of control messages throughout the network and therefore saves bandwidth. Several productions currently employ OLSR [73, 20, 96, 75, 4]. OLSR performance increases when there are multiple possible paths from source to destination. The performance achieved by OLSR is higher in networks where there is a greater possibility of using alternative routes through information traffic.

To reduce control message overhead, each OLSR node selects, among its one-hop neighbors, a subset of nodes to retransmit this control messages. Therefore, only nodes in this subset, called MPR, forward control messages of a given node. Thus, nodes in the MPR set must reach all two-hop neighbors to deliver messages to all nodes in the network. With OLSR, a node starts selecting as MPR the node whose cover a greater number of two-hop neighbors. The goal is to decrease the amount of control message retransmissions. In addition, nodes belonging to a set of MPRs must have symmetric links between the nodes whose selected them. This symmetric links will allow an efficient packet forwarding.

Figure A.1 illustrates how node A defines its MPR set. There are a set of one-hop neighbors that are in the same coverage area of node A. Those neighbors nodes are: B, C, D, E, F. To reach two-hop neighbors, node A will have to use other nodes to forward the messages. D and E are selected as MPRs because both are the ones that reach as many two-hop neighbors as possible.

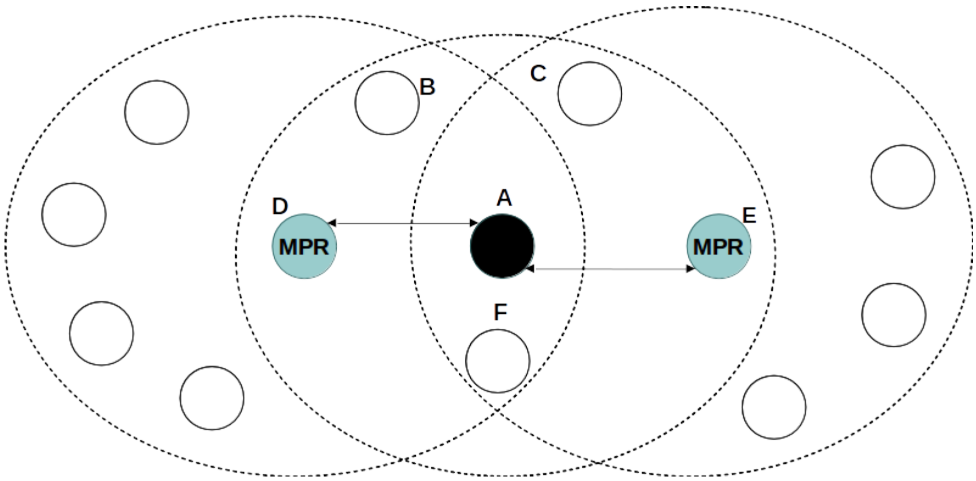


Figure A.1: MPR Selection.

The MPR selection technique significantly reduces the number of retransmissions needed to keep all network nodes up to date. OLSR uses MPRs nodes to filter the amount of link state control messages distributed on the network. This reduction occurs because the MPRs nodes declare their link state information to other nodes that use the same MPR and if an additional topological information is needed, they will serve for redundancy purposes. Because MPRs reduce the flood of control messages on the network, it is possible to optimize the reactions to OLSR topological changes by decreasing the OLSR refresh interval time without excessively compromising network bandwidth.

The information between nodes and MPRs is exchanged through probing packets, also called hello packets. The hello packets are sent periodically between the nodes, and when nodes have multiple interfaces, each interface is responsible for sending its own hello packets. The set of MPRs is not fixed and can be changed as the information obtained with the hello packet indicates link changes or an inoperative node. If the link layer provides sufficient information in its transmissions, it can be used by OLSR to check the state of the links. This check works by listening for exchanged messages among nodes, thus reducing the number of hello messages sent.

If it is necessary to use broadcast for all nodes of the network without limiting the use of the MPRs, OLSR has a message type that is sent to near nodes and will be retransmitted to all its neighbors and so on until the entire network is quickly reached.

The control messages do not have to be delivered in an orderly manner to the routers that receive them since each control message has a sequence number that is incremented at each new message. Therefore, the node that receives a control message can identify and use the messages that have the latest information.

To create a topological information base, each node chosen as MPR sends network topology control messages called Topology Control (TC). TC messages have to reach all nodes of the network. Thus, MPRs broadcast the TC messages on the network providing to each node a sufficient link state information to calculate its routes.

OLSR is designed to work in a fully distributed way and thus does not require the use of central control entity in the network. Furthermore, it is not necessary that there is reliability in the state of the links, the reason for this is because the protocol is constantly checking their status through control messages. Consequently, it is possible to react to a link that is having a high number of losses by looking for another link that satisfies the needs of the traffic.

Due to the aforementioned OLSR characteristics, this protocol was chosen to make the comparative tests with CCN. Despite the existence of many other routing protocols for wireless mesh networks, OLSR was also chosen for its wide use in academia [44, 37, 3, 68, 60, 51, 41, 2]. Combined with this protocol we use the ETX metric, described below.

A.1.1 ETX Metric

Routing metrics play a key role in the performance of routing protocols. In many comparative metric studies, it can be observed that changing the metric with the same protocol can change performance considerably [26]. The metric used in the comparative tests presented in this study is the Expected Transmission Count (ETX) [36]. The reason for choosing this metric is also based on the high popularity among other metrics on academics works.

ETX metric can be defined as the expected number of transmissions needed to deliver a packet through a given link [36]. This metric calculates the weight of the paths through the sum of all the ETXs of the links in a path [33]. Thus, the metric will choose paths that decrease the total number of retransmissions at the link level.

The probability of transmission success is calculated using probing packets. These packets are sent at regular intervals, which makes possible to create an estimate of the number of packets received at a given time. Some routing protocols have control packets sent at regular intervals. Therefore this packets can be used to calculate ETX [51].

Equation A.1 represents the probability of successful transmission between a node X and a node Y. Once a time interval has been set, the transmission probability calculation can be done with the number of received packets PR divided by the number of expected packets PE. A routing algorithm using the ETX metric will choose the paths that present the lowest sums from ETX values belonging to the links that constitute the path.

$$P_{xy} = PR/PE \tag{A.1}$$

ETX metric uses two variables to store the data related to probe results success, Equation A.2, where DL (Direct Link) and RL (Reverse Link) messages.

$$ETX = 1/\left(LD * LR\right) \tag{A.2}$$

It is important to note that the ETX metric calculates its LR by the number of packets received in a time interval. Therefore, each node periodically sends probe messages with route maintenance information. For example, considering an interval of 20 s and transmitting one probe message every 2 s for each neighbor nodes. Thus, a 100% quality link would have received ten packets in 20 s interval. LD calculation is done in the same way, but its value is informed by the neighbors. This information is collected when the neighbor nodes use the reverse information to their probe messages. Thus, each node indicates in the probe messages the number of packets received from each of its neighbors.

Comparisons results between the ETX metrics and the hop count metric show that ETX improves the network performance by choosing routes with less packet loss probability [87].However, the ETX metric may not accurately reflect the probability of these losses. This behavior occurs mainly because the metric use probe messages packets to infer the transmission probabilities and these packets have a reduced size compared to data packets. Thus, ETX metric can make a less realistic estimate than if it was using packets that are closest to the data packets to infer the probability of transmission. Besides, these packets are sent at a basic transmission rate (physical rate), which may provide imprecise estimates of network behavior about different transmission rates.

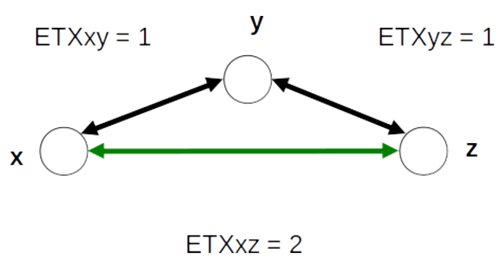


Figure A.2: Choosing the best nodes for MPRs.

Another problem found in the ETX metric is the possibility to settle routes that can degrade network performance[77]. This degradation can occur due to the number of packet losses, exemplified in Figure A.2. The link between nodes x and y has an ETX value of 1 and the link between n and y also has an ETX value of 1. Thus, using the equation of probability, Equation A.1, it is noticed that an ETX value equal to 1 means that the same number of expected packets were received, i.e., the link has a 100% transmission success probability. For the link between the nodes x and z, it is showed that ETX value is equal to 2, which represents a transmission success probability of 50%, using Equation A.1. The problem appears in the calculation of the ETX value between the nodes x and z because there are two routes one direct and another through the y node. The two routes have the same ETX value since the sum of the xy and yz will be 2. Therefore, ETX metric will choose the route xz, because it has the same ETX value of the other route, xyz, but with fewer hops. However, route xz has a higher loss rate and consequently will affect network performance.