Complementing IoT Services Using Software Defined Information Centric Networks: A Comprehensive Survey

Wajid Rafique, Abdelhakim Senhaji Hafid, Soumaya Cherkaoui

Abstract—IoT connects a large number of physical objects with the Internet that capture and exchange real-time information for service provisioning. Traditional network management schemes face challenges to manage vast amounts of network traffic generated by IoT services. Software-Defined Networking (SDN) and Information-Centric Networking (ICN) are two complementary technologies that could be integrated to solve the challenges of different aspects of IoT service provisioning. ICN offers a clean-slate design to accommodate continuously increasing network traffic by considering content as a network primitive. It provides a novel solution for information propagation and delivery for large-scale IoT services. On the other hand, SDN allocates overall network management responsibilities to the central controller, where network elements act merely as traffic forwarding components. An SDN-enabled network flexibly supports ICN without deploying ICN-capable hardware. Therefore, the integration of SDN and ICN provides benefits for large-scale IoT services. This paper provides a comprehensive survey on Software-Defined Information-Centric Internet of Things (SDIC-IoT) for IoT services provisioning. We present critical enabling technologies of SDIC-IoT, discuss its architecture, and elaborate its benefits for IoT services provisioning. We elaborate on key IoT service provisioning requirements and discuss how SDIC-IoT supports different aspects of IoT services. We develop different taxonomies of SDIC-IoT literature based on various performance parameters. Furthermore, an extensive discussion on different use cases, synergies, and advances is presented to envision the SDIC-IoT concept. Finally, we provide current challenges, causes, and future research directions of IoT services provisioning using SDIC-IoT.

Index Terms—Internet of things, Information-centric networking, Software-defined networking, Service provisioning

I. INTRODUCTION

Internet of Things (IoT) aims to connect physical objects with the Internet, providing any time, anywhere, and any path connectivity [1]. These connected physical objects include smartwatches, health meters, temperature sensors, proximity sensors, accelerometers, and optical sensors. The connectivity of these objects gives rise to novel real-life services over a wide range of fields, including healthcare, smart cities, smart manufacturing, and industrial control systems. IoT can be

deployed with sensing and intelligent capabilities to communicate, collaborate, and exchange real-time data for autonomous control of infrastructures. Diverse IoT applications and services require higher transmission capability than ever before. Services provided by IoT vary according to their use cases, but the infrastructure used for communication is shared across all networks. The current Internet model was designed to connect different network endpoints many decades ago [2]. Most of the Internet communication is performed between clients and servers using unique IP addresses. However, the majority of IoT services rely on content sharing between the content providers and consumers rather than interacting with particular servers [3]. The sensory infrastructure is highly mobile, and the data dissemination from IoT devices depends on different wireless networking technologies, including cellular, WiFi, Low Power Wide Area Networks (LPWANs), Bluetooth low energy, Zigbee, and WLAN [4]. Hence, different add-ons are continuously developed in the existing protocol stack to support novel network protocols [5].

The connectivity of IoT devices produces a massive amount of data giving rise to real-time big data streams. The access, discovery, and analysis of data put more constraints on the current TCP/IP architecture underpinning many vital issues [6]. One of the main concerns is the naming of IoT devices. Currently, the IPv4 naming space is exhausted, further portraying threats on the IPv6 address space that could be exhausted in the future [7]. Another challenge arises due to the long address length of IPv6, which is challenging to implement on constrained IoT devices. In this regard, efficient naming and addressing schemes for IoT devices are not duly available in this infrastructure [8]. Moreover, IoT suffers from different constraints in battery power, processing, size, usage (in-body/out of the body), memory, and storage. IoT devices comprise heterogeneous hardware and vendor-specific configurations requiring device-specific operating systems. Due to the limited size, battery, and storage constraints, data often become unavailable, causing data unavailability issues [9]. To cater to these challenges, solutions like in-network caching to make data available tend to be missing in the traditional IP-based networks [10]. Moreover, IoT services require more security and extra privacy based on data accessed and its usage [11]. Furthermore, IoT services require data mobility, which suffers from mobility management, fault-tolerance, and hand-off issues. IoT requires network architectures that facilitate changing network paths, provide technological abstractions, and consider data exchanges instead of access nodes [12].

W. Rafique and A. S. Hafid are with the Department of Computer Science and Operations Research, University of Montreal, Montreal, QC H3C 3J7, Canada (e-mail: ahafid@iro.umontreal.ca (e-mail: rafiq.wajid@umontreal.ca)).

S. Cherkaoui is with the Department of Electrical and Computer Engineering, Universit[Pleaseinsert"PrerenderUnicode"~intopreamble] de Sherbrooke, Sherbrooke, QC J1K2R1, Canada (e-mail:soumaya.cherkaoui@usherbrooke.ca).

(Corresponding author: W. Rafique.)

Fig. 1. An example of IoT service provisioning using SDICN paradigm.

This increases network complexities making it challenging to implement novel network configurations [13]. In this context, Software-Defined Networking (SDN) and Information-Centric Networking (ICN) are proposed to address the limitations of current communication technologies at the design level [14].

ICN has proved to be a novel communication model for the content distribution challenges faced by IoT [15]. It matches a wide range of IoT applications that are information-centric as they focus on data regardless of the origin of the data [16]. It offers flexibility to address the content directly by name instead of the address of the content [6]. It introduces named content instead of IP packets that can be transmitted in the network towards the destination. This offers a flexible communication model shifting away from a host-centric environment to a content-centric paradigm [17]. In this case, the actual target is the content itself rather than the address of the content. Hence, in-network caching can be implemented by storing content near the user vicinity to improve data retrieval and reduce network congestion. ICN efficiently manages mobility by re-issuing any unsatisfied requests considering the fact that the content is independent of its location [18]. ICN provides instant data access using a request-reply exchange mechanism, content-based security, and multicast support that is specifically beneficial for IoT [19]. Although this seems to be conceptually simple; however, it becomes challenging to implement it as an overlay on existing networks [9]. ICN creates increased complexity of IP routing, which makes it hard to fully utilize its functionality in legacy networks [20].

Currently, Internet architecture faces challenges to scale due to the content-centric requirements of the underlying applications [28]. A vital aspect of the future Internet is that it supports a content-driven paradigm rather than a connection-driven. Internet was designed for the entity-based standpoint providing a communication platform through established connections [29]. A wide variety of content-centric IoT services cause communication overhead on the current Internet architecture [30]. Although the content-centric nature of IoT provides the utility of access, it causes discrepancies in the current Internet architecture [31]. Content-centric sys-

tems exchange content without considering the origin of the content. However, its implementation requires designing novel architectures based on various prioritization schemes for the content regardless of its origin. SDN becomes helpful in this situation by supporting programmability in the networks and implementing network functions flexibly.

SDN aims at increasing network programmability by separating control and data planes [1]. It develops a layered network architecture by dividing it into the application, control, and data planes. In addition to the layered architecture, multiple Application Programming Interfaces (APIs) exist, including northbound, southbound, and east/westbound interfaces supporting the flexible programmability of SDN. The northbound API provides an abstract view of the network to the application plane, whereas the southbound API provides a communication interface between data and control planes. This enables the controller to deploy traffic forwarding rules in the network elements. The east/westbound API coordinates with multiple controllers to present a logically centralized controller view and supports integrated decisions. Adequate control logic is implemented on the forwarding elements by the controller based on service requirements to forward network traffic. SDN provides flexibility to develop, deploy, and improve network services in an agile and flexible way.

A. Motivation of this Survey

Fig. 1 shows a use case of IoT service provisioning using SDN and ICN. It shows sensors connected to the physical objects capturing real-time data and delivering services to the intended stakeholders. The motivation of this survey is prompted by the need for SDN and ICN for IoT services and the lack of existing surveys on this concept. IoT devices can deliver content-based services where SDN and ICN could act as complementary technologies for IoT services delivery. These services overwhelm the network with huge amounts of content, posing network management challenges. To offer solutions to these challenges, several techniques use SDN and ICN for IoT service provisioning separately. However, SDN and ICN are promising paradigms that could benefit from each other to solve multiple networking challenges of IoT services. Therefore, researchers have already started to integrate these paradigms to develop a Software-Defined Information-Centric Internet of Things (SDIC-IoT) architecture for seamless service provisioning [32]–[34]. SDN benefits of centralized control, robust management, and a global view of the network also apply to ICN. SDN-enabled, ICN can handle the distribution of IoT data in a partially upgraded network, simplify the deployment, and consolidate the transition phase. Moreover, the ICN feature of in-network caching can efficiently handle information delivery from dead (unavailable) IoT devices by caching contents at intermediate nodes [35]. It minimizes delay caused by data retrieval in the case of alive devices using optimal caching strategies developed using the centralized control of SDN [36]. SDIC-IoT realizes an overall optimal resource allocation strategy using a centralized controller. Thus, IoT content could be cached at nearby nodes for efficient service provisioning. Furthermore, SDN could reap the benefits of

TABLE I
A COMPARISON OF THIS SURVEY BASED ON THE DISCUSSED CHARACTERISTICS WITH THE EXISTING SURVEYS.

Reference	Year	SDN	ICN	IoT	NFV	SDIC-IoT	Network Management	Scalability	Mobility	5G-6G Networks
Amadeo et al. [3]	2016	×	✓	✓	×	×	×	✓	✓	✓
Arshad et al. [16]	2018	×	✓	✓	×	×	×	✓	×	✓
Salman et al. [21]	2018	✓	×	✓	✓	×	✓	✓	×	×
Fang et al. [22]	2018	×	✓	✓	×	×	✓	✓	✓	×
Fang et al. [23]	2017	×	✓	✓	×	×	×	✓	×	×
Zhang et al. [24]	2015	×	✓	✓	×	×	✓	✓	✓	×
Bekri et al. [5]	2020	✓	×	✓	✓	×	✓	✓	×	✓
Farris et al. [25]	2018	✓	×	✓	✓	×	✓	✓	×	×
Bizanis et al. [26]	2016	✓	×	✓	✓	×	✓	✓	×	×
Zhang et al. [14]	2018	✓	✓	×	✓	✓	✓	✓	✓	×
Jmal et al. [27]	2017	✓	✓	×	×	×	✓	✓	×	×
This Survey	2021	✓	✓	✓	✓	✓	✓	✓	✓	✓

in-network caching and adapt to data-centric advantages of data-centric security and mobility for IoT services delivery. ICN changes the end-to-end communication paradigm having a profound impact on the design and functioning of SDN. For example, name-based IoT traffic forwarding at the data plane devices is more efficient than flow-based forwarding [37]. Moreover, data naming could resolve the address space scarcity issues of IPv4, enabling scalability and offering namespace management and data retrieval produced by IoT. Furthermore, mobility handling of ICN provides better content delivery for critical IoT mobile devices. The privacy issues in resource-limited IoT could be better solved by ICN allowing the security of data rather than securing hosts. Motivated by these benefits, we survey the existing literature on SDN and ICN for IoT service provisioning in this paper. This article provides background, motivation, and advantages of integrating SDIC-IoT. It provides a taxonomy of the existing literature on SDIC-IoT and presents the requirements for successfully implementing SDIC-IoT. It offers novel use cases that can benefit from a promising SDIC-IoT infrastructure and discusses current challenges and future research directions.

B. Comparison With Other Related Surveys

This survey is unique compared to already available surveys in a way that it includes a holistic approach for SDIC-IoT-based caching, naming, network management, and mobility handling. Although various surveys have been published for ICN and SDN, most discuss individual aspects of both paradigms. A thorough search of the existing literature shows that some surveys discuss ICN and SDN individually for IoT services [3]. Arshad et al. [16] present an extended discussion on ICN models, their feasibility, caching, and naming mechanisms for IoT. However, IoT data management becomes a critical challenge considering the scale and volume of data produced by IoT. In this regard, Salman et al. [21] present a survey for IoT service provisioning using SDN and fog computing. However, ICN caching can be beneficial to deliver huge amounts of data generated by IoT efficiently. Some existing surveys discuss ICN for IoT mobility [22], energy efficiency [23], and caching [24]. Similarly, some SDN surveys focus on traffic engineering [5], security [25], and virtualization [26] for IoT. There are a few surveys that discuss the integration of SDN and ICN [14], [27]. In this context, Zhang et al. [14] propose a survey on the integration of SDN

and ICN. This survey focuses on communication aspects of SDN and ICN for general-purpose computing. They survey ICN caching mechanisms based on SDN; however, their study lacks implementation schemes. Jmal et al. [27] survey Content-Centric Network (CCN) management using SDN. However, it has a limited scope and lacks comprehensive aspects of SDN and ICN integration. Furthermore, this study [27] did not cover the novel characteristics of Software-Defined Information-Centric Networking (SDICN) that could be beneficial for the resource-limited IoT. There is a lack of existing surveys that address the integration of SDN and ICN for IoT service provisioning. To the best of our knowledge, this survey is the only one that summarizes SDICN for IoT by considering core issues faced by IoT in TCP/IP networks for service provisioning. Table I shows the comparison of this survey with existing surveys based on various parameters that we address in this paper.

TABLE II
LIST OF ACRONYMS AND THEIR EXPLANATION.

API	Application Programming Interfaces
A-CPI	Application-Control Plane Interface
CCN	Content-Centric Networking
ComBo	Convergence of fixed and Mobile BrOadband access
C-DPI	Control-Data Plane Interface
DONA	Data-Oriented Network Architecture
DPI	Deep Packet Inspection
FIB	Forwarding Information Base
FIA	Future Internet Architecture
ICN	Information-Centric Networking
IETF	Internet Engineering Task Force
IoE	Internet of Everything
IIoT	Industrial Internet of Things
IoT	Internet of Things
IoMT	Internet of Medical Things
ISP	Internet Service Providers
IoBT	Internet of Battle Things
LPWANs	Low Power Wide Area Networks
M2M	Machine-to-Machine
NSF	National Science Foundation
NDN	Named Data Networking
NFN	Named Function Networking
NFV	Network Functional Virtualization
POINT	iP Over ICN the betTer IP
PURSUIT	Publish-Subscribe Internet Technology
POF	Protocol-Oblivious Forwarding
SAIL	Scalable and Adaptive Internet Solutions
SDN	Software-Defined Networking
SDIC-IoT	Software-Defined Information-Centric Internet of Things
SDICN	Software-Defined Information-Centric Networking
TRAID	Internet Architecture Integrating Active Directories
TCAM	Ternary Content Addressable Memory
V2V	Vehicle-to-Vehicle
V2I	Vehicle-to-Infrastructure
WoT	Web of Things
XIA	eXpressive Internet Architecture
5G	Fifth Generation

C. Contributions of this Survey

In this survey, we discuss the integration of SDN and ICN to solve communication, content management, and service provisioning challenges faced by IoT. We provide a comprehensive literature survey on the existing SDN and ICN-based approaches of in-network caching, content naming, and mobility handling mechanisms for IoT. This is a pioneering survey, which provides the following contributions.

- We discuss the underlying motivation of integrating SDN and ICN and the key benefits that it brings for IoT service provisioning.
- We explain and critically analyze the requirements of SDIC-IoT and its adaptability towards IoT for content naming, caching, security, and mobility handling.
- We categorize existing literature that focuses on integrating SDN and ICN and classify it based on different parameters, including mobility, virtualization, caching, and implementation paradigms.
- We discuss novel use cases of SDIC-IoT that provide a model for future SDIC-IoT systems.
- We present several open issues, challenges, and future research directions regarding the implementation of future SDIC-IoT systems.

D. Organization of this Survey

Table II shows acronyms and their description. This paper is organized as follows. Section II provides an overview of SDN, ICN, and IoT; it discusses the motivation and benefits of SDIC-IoT integration. Section III presents the requirements of seamless SDIC-IoT implementation. Section IV presents the taxonomies of existing approaches of SDIC-IoT. Section V presents novel uses case, synergies, and implementations of SDIC-IoT. Section VI presents current issues, challenges, and future research directions. Finally, Section VII concludes the paper.

II. SOFTWARE-DEFINED INFORMATION-CENTRIC INTERNET OF THINGS

In this section, we discuss the background of SDN, ICN, and IoT. We provide the introduction of all the technologies and establish the motivation and critical benefits of SDIC-IoT integration.

A. Information-Centric Networking

The prime advantage of ICN is its capability of addressing network content by names instead of host-centric addressing [6]. ICN is based on the publish/subscribe Internet architecture, which focuses on efficiently delivering user content regardless of the origin of the content. As compared to the host-centric connectivity in IP networks, ICN allows its connected users to obtain required content from any location in the network [16]. Therefore, ICN could be exploited to solve the challenges in current IP networks, including mobility, energy efficiency, congestion, and security [32].

1) *Evolution of ICN*: Fig. 2 shows the evolution of ICN since it was first introduced in 1999 by Cheriton while addressing name-based routing in Translating Relaying Internet Architecture Integrating Active Directories (TRIAD) architecture [38]. National Science Foundation (NSF) of the USA has proposed different projects that target future Internet Architectures, including Named Data Networking (NDN), Nebula, eXpressive Internet Architecture (XIA), and ChoiceNet [39]. UC Berkeley and the European Commission (EU) also introduced numerous novel projects that include Data-Oriented Network Architecture, COMET, CONVERGENCE, Scalable and Adaptive Internet Solutions (SAIL), and Publish-Subscribe Internet Technology (PURSUIT) [40], [41]. Subsequently, in 2007 the concept of (Data-Oriented Network Architecture) DONA was introduced as the first clean slate architecture of ICN [42]. In 2009, (Content-Centric Networking) CCN was proposed based on replacing IP protocol stack [43]. Later CoNet, Convergence of fixed and Mobile BrOadband access (COMBO), and Content-centric fashIon mANET (CHANET) adopted the concept of CCN [44]. Since the architecture of CHANET is fundamentally the same as of CCN; however, it has not been taken much into consideration [45]. Later in 2010, NSF introduced a novel program for Future Internet Architecture (FIA), which introduced MobilityFirst [46]. This program devised an architecture on mobility to offer seamless communication among users based on ICN principles. Another add-on of the ICN program was NDN that was an extension of the CCN approach. Similar projects related to this include GreenICN and Named Function Networking (NFN) [12]. However, NFN does provide specific differences as compared to NDN. As part of the FP7 European program, COMET proposed a content-oriented architecture based on CCN and NDN concepts for content access and dissemination in an efficient way [42]. In 2015, the EU funded two more projects, including iP Over IcN the betTer IP (POINT), and UMobile [47]. These programs were based on the assumption that some IP-based applications could benefit from ICN-based networks. POINT provides a platform, which supports IP-based applications such as legacy applications during the interim period of a globally deployed ICN network. UMobile focused on an ICN-based mobile-centric architecture to assist communication among mobile participants such as sensors, actuators, and mobile devices [48]. Both projects are in their evolutionary stage; therefore, the final version of this architecture is not yet available. This is also implemented in the ICN 2020 project that started in mid-2016 and discussed one of the goals to improve underlying ICN solutions in the IoT paradigm.

A robust architecture of ICN is CCN, which offers communication using two specialized packets that include Interest and data packets [20]. These packets use a name to address the desired content uniquely. The Interest packet issues a user's request to identify the underlying Interest data while the data packet returns the related content to the user. Among other proposals for the future Internet, ICN efficiently deals with the challenges faced by current IP-based networks [49]. It aims at ensuring efficient data dissemination and integration of the overall network features around the content. The fundamental element in ICN is the content name, which has the same

Fig. 2. Timeline of ICN evolution over the past few years.

significance as IP having unique, persistent, secure, location-independent, and friendly nature in the overall network context. Considering the content-oriented nature of ICN, content names could be represented in various ways, such as attribute-value, flat, or hierarchical. Therefore, the routing plane of traditional IP-based networks could be modified by deploying host addresses to utilize the content names for name-based routing. Moreover, in-network caching and content replication allow data access and retrieval during communication.

B. Internet of Things

IoT refers to billions of devices connected to the Internet that collect and share data; it impacts society in various ways. The sensing capability, along with the availability of wireless sensor networks, turns these objects into IoT. Things in IoT can be humans and other objects connected using Machine-to-Machine (M2M), machine-to-human, and human-to-human interfaces. Many definitions exist in the literature for IoT. IETF defines IoT as uniquely identifiable objects connected using TCP/IP and non-TCP/IP suites [50]. IEEE defines IoT as uniquely identifiable physical and virtual objects employing existing, and novel communication protocols [51]. IoT is an umbrella term having underlying use cases including M2M, Industrial Internet of Things (IIoT), Social Internet of Things (SIoT), Web of Things (WoT), Internet of Medical Things (IoMT), and Internet of Everything (IoE). M2M is a use case of IoT, which deals with autonomous communication among machines as well as humans to machines [3]. IIoT enables communication among industrial objects for industrial service provisioning [15]. SIoT offers communication among IoT, and online social networks [52]. Furthermore, IoMT develops an autonomous architecture to connect medical objects for healthcare service provisioning [53]. IoE establishes a communication paradigm among humans, machines, and all other objects connected to the Internet [54]. WoT pushes data collected by IoT objects to the web to provide innovative applications [55].

Users of IoT applications are more interested in getting access to content instead of the address of the content source. This is because of the vast proliferation of content-oriented applications during the past few years. For example, every device in IoT collects data at a larger and continuous scale. TCP/IP was initially designed to serve the expensive network resources using a limited address space [56]. Evidentially, it was not originally intended to serve the purpose of billions of uniquely identifiable IoT objects. Moreover, the sophisticated nature of IoT puts vast constraints on content dissemination

TABLE III
THE UNIQUE CHALLENGES IN IoT SERVICES PROVISIONING.

Parameters	Challenges	SDICN
Data collection	Huge traffic	Content caching reduces network traffic
Continuous data streams	Device naming issues	Content-oriented forwarding
Limited processing power	Security and privacy	Network level security solutions
Lack of standardization	Interoperability issues	Centralized management
Lack of power	Device unavailability	Caching content on intermediate nodes
Heterogeneous infrastructure	Different communication parameters	SDN based interoperability schemes
Billions of devices	Scalability issues	Distributed controllers
Data transfer	Communication among untrusted entities	SDICN integrity enforcement

and scalability of the underlying architecture. Therefore, ICN has emerged as an ideal candidate to fulfilling the needs of IoT infrastructure [57]. The primary characteristics of ICN include in-network caching, naming, security, mobility, and scalable information delivery that are highly suitable for IoT services [8]. Furthermore, ICN-oriented hourglass framework supports thin-waist like TCP/IP [58]. Additionally, ICN could mask over the traditional TCP/IP layer or MAC layer. Although the existing literature [46], [59] argue that ICN seems to be replacing TCP/IP, whereas ICN is an overlay network that sits on an IP network. To be precise, CCN is a layer that envelops the need for relating content with IP addresses rather than names. However, the actual content delivery needs TCP/IP interface or direct MAC (layer 2) interface [59].

C. Software-Defined Networking

Communication networks weave the fabric of the current connected world. SDN is a novel paradigm of network management aiming at separating data and control planes to manage network components flexibly [28]. The controller in SDN acts as the brain of the whole network performing most of the computations. The Data plane contains forwarding devices (e.g., routers and switches) responsible for flow rule matching to forward incoming packets. Forwarding devices send packets to the controller if the relevant flow rules are not

Fig. 3. An architecture of software-defined information-centric network for IoT showing IoT devices at the data plane. The modified forwarding devices at the data plane act as caching elements considering the resource demands of IoT services.

available in the limited Ternary Content Addressable Memory (TCAM) memories. The controller analyzes the flow and installs the corresponding flow rule in the data plane device, which then forwards the packet towards its destination. SDN brings independence between control and data planes, offering flexibility to the service providers and application developers to proactively manage network resources and support network expansion [28]. It provides flexible network evolution where network operators configure, upgrade, and maintain network resources dynamically.

SDN breaks the vertical architecture of network administration and separates the data and control planes for flexible network management. Thus, network operators and application developers can configure, upgrade, deploy, and flexibly maintain network services. The logical centralization of the network enables the controller to gain access to all network resources where traffic can be efficiently managed. Traditional networks handle packets using MAC addresses, IP prefixes, and combinations of IP and TCP/UDP port numbers using Control-Data plane Interface (C-DPI) [60].

1) *Application Plane*: The application plane establishes a platform for the application developers to create novel applications for desired traffic forwarding. It uses RESTful API to communicate with the control plane. Developers design novel applications for customized traffic forwarding, network policy implementation, and security enforcement. These applications can be implemented at the SDN controller for network-wide deployment.

2) *Control Plane*: The control plane is the core of SDN consisting of one or more software implementations of the controller. It efficiently allows communication from the controller to another, application plane to the controller and data plane to the controller. The control logic of the controller maps the specifications of network management to the flow rules for the forwarding components at the data plane.

3) *Data Plane*: The lowest plane in the SDN hierarchy is the data plane consisting of forwarding components, including switches, routers, access points, and connected objects. The central controller performs all the forwarding decisions

based on network topology, making network elements merely forwarding components. IoT devices are attached to data plane forwarding devices to connect to the backbone network. The central SDN controller uses OpenFlow to communicate with traffic forwarding components.

D. Software-Defined Information-Centric Networking For IoT

Table III shows challenges that are unique for IoT services provisioning. This table also shows SDICN characteristics that could be beneficial in addressing these challenges. A practical realization of ICN for IoT requires an efficient architecture hiding all the network complexities. In SDN, the centralized controller contains the network's global view, enabling efficient network resource management and control. SDN becomes a key candidate in this paradigm, providing greater network programmability by separating control and data planes. The control plane of SDN lowers down the implementation and architectural complexities of ICN, allowing novel mechanisms for resource management [9]. Although SDN simplifies network management, its architecture is still based on TCP/IP. Bringing ICN into SDN alleviates the challenges of host-centric networking. Shifting from 'where' to 'what' brings novel opportunities for efficient traffic management. SDN can optimally exploit in-network caching and multipath routing using name-based content retrieval with the global view of the network.

SDN uses a flow-based communication model that has several limitations. For example, flow-based forwarding performs the same treatment even if the flows' port number, source address, and destination addresses are identical. ICN provides the capability to inspect packet metadata without Deep Packet Inspection (DPI) [34]. The packet metadata describes the content length and type, enabling network service providers to optimize network behaviours accordingly. OpenFlow channel and its north and southbound interfaces are prone to attacks [61]. In this case, ICN can protect the content itself, securing the overall network. The data centre security model of ICN could protect the overall network enhancing efficient access to data generated by IoT. Both SDN and ICN complement

Fig. 4. A SDN caching strategy for IoT services where caching nodes are selected based on optimal paths.

each other, paving the way for efficient IoT services delivery bringing the novel idea of Software-Defined Information-Centric Internet of Things (SDIC-IoT). Fig. 3 illustrates the proposed architecture of SDIC-IoT where IoT devices are connected at the data plane. These devices interact with data plane components to transfer the captured data. Forwarding switches at the data plane act as caching components to store data near content consumers. The controller in this architecture determines optimal caching strategies based on the central view of the connected components. The modified application plane in the SDIC-IoT architecture contains ICN-specific applications, including routing, caching, and performance optimization. IoT devices communicate with data plane devices for content transfer. Upon receiving a content request, the content is forwarded to the controller, which installs relevant flow rules at the data plane devices. Data plane devices then deliver the content to the content consumers based on the flow rule. In this architecture, in-network caching and multi-casting characteristics reduce overall transmission time and increase network throughput. Directly identifying services by content names overcomes the domain-specific heterogeneity of IoT and provides an abstraction to deliver the content between content producers and consumers. Fig. 4 shows the content caching mechanism using SDIC-IoT architecture. A consumer generates a request for content C2. This request is directed to the caching node storing the content C2. This content is then forwarded to the consumer. Hence, this architecture efficiently manages network resources by keeping the content on fewer nodes. In the following, we present the benefits of SDIC-IoT.

1) *IoT Content Management*: ICN tries to solve the problems of content distribution efficiency, security, network congestion (exacerbated by the increase of content generated by IoT and shift in multimedia technology usage). At the same time, SDN brings flexibility in network management. The motivation for using the SDIC-IoT architecture comes from

the large-scale data generation capabilities of IoT. SDIC-IoT provides the flexibility to cache content on important switches that are frequently traversed by the network traffic. Hence, this architecture reduces frequent content request redirection and reduces latency in accessing the content. In traditional caching, content unavailability, replication, and request redirections are significant challenges. In traditional caching, content becomes unavailable when content gets no-hit in a specific time interval. Hence, it causes content redundancy and unavailability at the same time. Thus, it results in inefficient utilization of network resources and increases network management costs. SDN guides network traffic efficiently, having the global view of the network, which supports ICN capability of caching and routing simultaneously. Routing strategies can be flexibly configured using well-defined interfaces offered by SDN (e.g., OpenFlow, RESTful, and east/westbound interfaces). SDN also provides various types of customized services using different namespaces supported by ICN. Fig. 5 shows a taxonomy of caching management schemes. It shows how centralized caching using SDN overcomes the challenges of distributed caching strategies.

2) *Content-oriented Communication Model for IoT*: SDN simplifies network management by providing a global view of the network. However, the current SDN architecture is still based on TCP/IP. Therefore, the challenges of TCP/IP still exist in SDN. In this case, ICN solves the challenges of the host-centric paradigm of TCP/IP by relying on content names rather than the content location. Furthermore, IoT service provisioning relies on content transfer from source to destination. This content can be obtained directly from producers or through intermediate nodes storing the content. Due to the content-oriented model of ICN, SDN can efficiently utilize in-network caching and multipath routing to transfer IoT-generated content. In-network caching and multicast routing increase network throughput by decreasing overall data transmission throughout the network. Moreover, the SDN controller can find optimal routes for data caching and transfer for IoT services.

3) *Centralized Control for IoT Services*: The centralized control of SDN gives robust control of route selection and caching nodes management. Network operators can program networks (via A-CPI) using optimal route selection strategies supported by the controller. SDN provides content using different namespaces for different domains of IoT applications. Novel security services can be provided by key distribution and revocation, strengthening security in the resource-limited IoT. Evolving caching mechanisms of ICN for optimal IoT services can be efficiently managed using dynamic control supported by SDN. Various implementations of A-CPI and C-DPI support programmability to route traffic according to the user needs. Programmable control and data plane support the development of customized functionalities on specific network nodes. IoT services can benefit through centralized control of general-purpose networking devices by flexibly forwarding name-based traffic. ICN mechanisms of in-network caching suffer from content replication and proximity of caching decisions. In this case, SDN manages content caching status through a centralized view of the caching nodes. Moreover, SDN

Fig. 5. A difference between centralized and decentralized caching strategies.

always has the topology information of the network, which helps optimize resource allocation. Indeed, SDN on-path and off-path caching schemes reduce caching redundancies and increase the overall cache hit ratio.

4) *Energy Efficiency*: Resource-limited IoT devices have severe limitations on power and computing capabilities. Most IoT devices spend a significant part of their life in sleep mode and only awake when they require data exchange. Existing IoT networks handle energy at the medium access layer or above the transport layer. ICN caching coupled with SDN management capabilities reduces extra energy consumption. A service request could be accomplished by another node caching the content when the producer is idle. Having a local copy of the content reduces the extra energy consumption required to communicate with the producer.

E. Summary and Insights

This section discusses the architecture of SDIC-IoT supporting large-scale data generated by IoT through centralized network management. The SDIC-IoT architecture provides a promising solution for IoT service provisioning. However, it creates other challenges of fault-tolerance, privacy, security, and scalability of the heterogeneous SDIC-IoT infrastructure. Caching strategies can be effectively implemented at the edge nodes. It reduces network traffic to and from the central cloud and increases throughput. A hybrid flow rule installation strategy must be used to install flow rules on the data plane devices to fulfil the installation needs of the large-scale flow rule. The application plane in SDIC-IoT could be utilized to forward network traffic according to the stakeholders' requirements. The OpenFlow channel in this architecture may become a bottleneck due to a centralized control [62]. Although distributed control strategy could be exploited to address this challenge, it brings increased latency, lower throughput, controller-state synchronization, and security challenges. Hence, a practical realization of SDIC-IoT should consider the requirements of all the involved infrastructures, communication challenges, and resource allocation concerns.

III. REQUIREMENTS OF FUTURE SOFTWARE-DEFINED INFORMATION-CENTRIC INTERNET OF THINGS

Fig. 6 shows the taxonomy of SDIC-IoT services provisioning goals. These goals are further divided into six dimensions, with each having its requirements. It is crucial to achieving these goals for seamless IoT services provisioning. The deployment of SDIC-IoT must consider IoT capabilities of massive content generation and communication requirements at the design level. SDIC-IoT offers tremendous opportunities for IoT content management by leveraging ICN capabilities of independent data plane and SDN benefits of central control of data plane elements. However, this complex resource integration of IoT, SDN, and ICN on a single platform requires the consideration of heterogeneous domain-specific requirements. Table IV shows a taxonomy of requirements for the implementation of future SDIC-IoT systems.

A. Requirements of Heterogeneous Network Management

Continuous growth and applications of IoT produce huge amounts of content requiring latency-aware and efficient processing. An important aspect is the management of huge amounts of devices and content produced by them. Thus, suitable communication techniques are mandatory to distribute efficiently and optimally control network traffic. Traffic management involves techniques for load balancing and delay reduction. These requirements can be satisfied by SDIC-IoT using hybrid capabilities to manage underlying networks. In this situation, SDN provides a global view of the network to route traffic towards its destination optimally. ICN facilitates unique, persistent, and location-independent naming that IoT applications could directly address. Thus, SDICN can be used for IoT network management for load balancing, reducing delays, optimized traffic forwarding, and efficient bandwidth utilization. Moreover, data analytics techniques could be developed at the network's edge to reduce network traffic and simplify network management.

However, centralized management of SDN poses significant challenges to the implementation of SDIC-IoT. Therefore, network management requirements ask for efficient management of SDIC-IoT using efficient strategies that have been tested on large scale networks [63], [64]. Such as the strategy mentioned in [63] prevents the controller from becoming a single point of failure using different management and load balancing strategies. The authors in [64] proposed a framework for topology and device management to fulfil application-specific requirements of IoT. This strategy uses network management policies, which modifies network topologies at run time to meet application-specific requirements of IoT.

B. Network Function Virtualization Requirements

Traditional networks rely on the predefined programmability of network devices, making it difficult to program them for multitasking [65]. Virtualization of network devices supports network function update in real-time, allowing them to perform multitasking. Implementation of Network Functional Virtualization (NFV) for the Internet Service Providers (ISP)

Fig. 6. A taxonomy of IoT services provisioning goals.

becomes easier due to the separation of control and data plane devices. Furthermore, SDN allows optimal network function placement for guiding flexible traffic flow. Function-centric service chaining extends ICN by transferring content naming to function names [66]. It separates network function from its location. It improves the scalability and flexibility of service chaining. Combining NFV with SDN enables novel application development through agile network resource management [33]. ICN establishes a novel way of network management by data plane programmability offered by SDN and NFV.

C. Requirements of Location-Independent Information Access

Device owners connected using IoT need to access their data independent of the location. In this regard, SDIC-IoT can control the devices and flexibly change network functions independent of location. An information access service must allow efficient content access based on content name, context, and information metadata. The requirements for such services should allow scalability, fault-tolerance, and latency of content access. SDN provides these capabilities by flexibly controlling the data plane using C-DPI. An API could be developed for efficient content access by hiding the complexities of the underlying hardware and providing information advertising and lookup. However, the requirements of such a service to identify the capabilities of end devices should be proactively considered before developing such an API. Although IoT may implement the API or use gateways, security issues arise due to insecure transmission mediums. Therefore, security and privacy requirements should be considered for seamless information access on an attack-vulnerable network.

D. Security and Privacy Requirements

IoT establishes interoperability among different cyber-physical systems for service provisioning. The resource-limited nature of these devices exposes them to different types of security and privacy attacks. Resource limitations and the wide connectivity of IoT make it a prime target for adversaries. One of the examples of attacks using IoT is the Mirai botnet attack. In these attacks, the adversaries blocked access to vital servers and ISPs in the US using

Mirai malware [67]. IP doesn't contain security by design, which was later included to support authentication and data integrity [68]. In this case, wireless communication among low-powered nodes degrades the performance of existing protocols. IP-based security protocols (IKEv2/IPsec, TLS, and EAP, etc.), IPv6 over 6LoWPAN, and CORE vastly depend on the location of nodes in the network [69]. However, these protocols secure the communication channel rather than the content. When security is ensured with other capabilities, such as mobility, it creates more network overhead, and complex management [70]. SDIC-IoT supports secure content sharing among nodes since authentication and integrity can be ensured locally. It eliminates trust issues that arise due to the content dissemination among intermediary nodes. The security of content reduces the overhead of securing the whole communication medium.

Furthermore, the programmable nature of SDIC-IoT could be used to develop security applications, protocols, and waypoint enforcement schemes [71]. These applications can be implemented at the application plane to enforce security policies in SDIC-IoT restricting content access to malicious nodes.

E. Interoperability Requirements Among SDN, ICN, and IoT Infrastructure

IoT provides services in all the significant fields of life, which require interoperability among different ecosystems, technologies, and protocols. IoT traffic passes through other network infrastructures arising interoperability issues. Network owners implement domain-specific policies, including data format restrictions, security strategies, and waypoint enforcement schemes [62]. These policies exacerbate interoperability issues due to the lack of a centralized governing authority. Furthermore, interoperability is sought for the heterogeneous IoT infrastructure to operate seamlessly among heterogeneous network entities. ICN characteristics of content-based addressing independent of location enhance the interoperability. Thus, the controller could be programmed to forward traffic based on content names rather than the content addresses [72]. Interoperability gateways can be deployed between different network domains that may perform content translations requiring unified interoperability mechanisms. Southbound OpenFlow interface of SDN provides a standard way to address hardware elements attached to the network. These elements merely act as forwarding components, thereby alleviating any hardware dependencies in addressing the content. However, flow rule installation strategies, limited TCAM memory management, and flow setup latency aggravate interoperability challenges. Additionally, new protocols and virtualization standards can be developed to support interoperability.

F. Mobility Requirements of SDIC-IoT

Mobility has become a salient feature of IoT due to device transformation into smart gadgets producing and transmitting data in real-time. Mobility can be of two types, including consumer mobility and producer mobility [73]. ICN supports consumer mobility by Interest re-transmission and in-network

TABLE IV

A TAXONOMY OF IoT SERVICES PROVISIONING REQUIREMENTS USING FUTURE SDIC-IoT SYSTEMS, THEIR CHALLENGES AND PROPOSED SOLUTIONS.

Requirements	Challenges	Solutions
Requirements of heterogeneous network management	<ul style="list-style-type: none"> • Huge amount of data production by IoT services. • Continuous growth of the network. • Lack of data analytics techniques. • Distribution and control of traffic. 	<ul style="list-style-type: none"> • Hybrid capabilities of SDN and ICN • SDN Capability of centralized control. • Location-independent addressing of ICN. • Load balancing delay reduction, and optimized traffic forwarding.
Network function virtualization requirements	<ul style="list-style-type: none"> • Predefined programmability of the network devices. • Lack of centralized view of the network. • Lack of agile network control. • Network function dependence on location. 	<ul style="list-style-type: none"> • Implementation of NFV for ISP. • Decoupling of control and data planes. • NFV-based network function placement. • Agile network resource placement.
Requirements of location-independent information access	<ul style="list-style-type: none"> • Location-dependent information access. • capability, fault-tolerance, and latency issues with content naming. • Lack of name-based addressing at data plane. • Lack of capability-identification at data plane. 	<ul style="list-style-type: none"> • Location-independent data access. • Flexibly changing network functions. • Development of name-based addressing at data plane. • Seamless API implementation at end devices.
Security and privacy requirements	<ul style="list-style-type: none"> • Heterogeneous infrastructures operating on a single platform. • Lack of security in IP by design. • Complexity in securing mobile devices. • Lack of content-protection schemes. 	<ul style="list-style-type: none"> • Local security policies implementation. • Lightweight security and privacy solutions. • Content level security solutions. • Security policies for heterogeneous infrastructures.
Interoperability requirements among SDN, ICN, and IoT Infrastructure	<ul style="list-style-type: none"> • Diversity of infrastructures. • Lack of centralized governance. • Lack of content translation among different networks. • Resource management issues at data plane. 	<ul style="list-style-type: none"> • Centralized governance of the network. • Content-based addressing of traffic. • Development of interoperability gateways. • Global communication protocol • Virtualization of hardware elements.
Mobility requirements of SDIC-IoT	<ul style="list-style-type: none"> • Large-scale development of mobile IoT. • Challenges associated with producer mobility. • Latency issues during network convergence. • Lack of seamless maneuvering of running services. 	<ul style="list-style-type: none"> • Interoperability using Interest re-transmission. • Optimal node selection for in-network caching. • Seamless maneuvering of services. • Solutions for producer mobility.
Scalability requirements of SDIC-IoT	<ul style="list-style-type: none"> • Huge traffic to and from Service providers. • Addressing of large-scale IoT objects. • Over flow of TCAM memory due to huge number of contents. • Controller bottleneck for cache status-awareness and latency-aware cache update 	<ul style="list-style-type: none"> • Intelligent content caching strategies. • Virtual implementations of single controller. • Optimizing east-west bound API of SDN. • Content scope definition using content names.
Requirements for fault-tolerant SDIC-IoT	<ul style="list-style-type: none"> • Asymmetric communication between IoT and communication infrastructure. • Highly connected nature of IoT. • Lack of backup paths. • Reactive backup solutions. 	<ul style="list-style-type: none"> • SDN capability of routing traffic. • Proactive fault-tolerant solutions. • Pre-programmed fault-recovery strategy. • Store and supply caching using ICN.

caching [74]. In a mobility situation, the data consumer requests the required data while moving to another location far from the caching router. In this case, the consumer re-transmits Interest for the same content. The intermediate router responds to the request. If data is not available at any caching node, the Interest calls the data producer, which delivers the content with a longer delay. However, there are still challenges when the data producer moves to another location, and the Interest is still not delivered [75]. The new name prefix of the producer might be unknown to any ICN router until the network converges. This halts the communication for at least the sum of handover time and the time taken for network reconvergence. In this case, SDN provides centralized control to efficiently select the closest router to cache packets based on consumer/producer mobility. However, IoT service request handling in a mobile environment without disturbing the running services is challenging. Therefore, seamless content delivery to the routers without worrying about the services is crucial to enhancing QoS in SDIC-IoT.

G. Scalability Requirements of SDIC-IoT

One of the prime requirements of IoT service provisioning is scalability. IoT promises to connect billions of smart objects, instigating the need for scalable networks capable of supporting large-scale infrastructure. Although current solutions such as IPv6 has a greater address space capable of addressing IoT devices, large-scale production of IoT poses future addressing challenges [16], [76]. Another challenge is the enormous

amounts of data produced by IoT, requiring efficient scalability management. Both SDN and ICN face challenges of scalability in different ways. IoT services generate a huge amount of content. In this case, the amount of content might be greater than the capacity of network devices. Thus, managing this content might provoke state space explosion. Establishing content forwarding at the data plane of SDIC-IoT based on content names might overflow limited TCAM memory of forwarding devices. Although the SDN controller provides the capability of state-awareness, cache status and latency-aware cache update aggravate efficient IoT service provisioning. These additional requirements cause scalability issues of the SDN controller. Moreover, flow setup latency based on content names at the data plane devices also causes major scalability issues [77]. Centralized management of SDIC-IoT poses restrictions on the single controller. Therefore, the architecture of SDIC-IoT should be scalable enough to support network content efficiently. Multiple distributed virtual instances of the controllers could be implemented to overcome controller scalability issues. In this case, the east-west bound API of SDIC-IoT could be utilized to provide communication among distributed controllers. In this way, a singular view of the controller could be presented to the data plane infrastructure. The content naming strategy of SDIC-IoT defines the scope of the content allowing users to retrieve the precise information required. The ability of ICN to transmit data to multiple users using anycasting and multicasting can alleviate scalability issues.

H. Requirements for Fault-Tolerant SDIC-IoT

Even in the failure of some of the SDIC-IoT components, the ability to provide services is essential to establishing fault tolerance. It involves fault detection, damage assessment, error recovery, and fault-treatment [78]. IoT relies on continuous Internet connectivity to provide services to the underlying systems. Fault tolerance in IoT services becomes challenging due to the asymmetric communication between IoT and communication infrastructure. SDN can route traffic towards available data plane devices even if some parts of the network fail. Faults in IoT are particularly probable due to the highly connected nature of IoT. Therefore, proactive fault-tolerant approaches should be implemented so that the system may recover before faults. SDN techniques could provide device backup paths providing the capability for the network to use them in case of a failure. The programmed strategy is activated in case of faults, and the traffic is routed towards the backup links. Backup controllers could also be implemented for fault recovery. The SDIC-IoT controller could be programmed to self-heal from different kinds of attacks. Store and supply mechanisms for content caching could be used to ensure end-to-end delivery of content. Moreover, to further establish fault tolerance, SDIC-IoT can cache content on intermediate nodes if some of the caching nodes become unavailable.

I. Lessons Learned: Summary and Insights

In this section, we discussed various requirements of implementing SDIC-IoT for end-to-end IoT service provisioning. SDN capabilities of programmability, centralized control, and flexible innovation could be beneficial for efficient ICN implementation. Combined SDN and ICN characteristics can provide a suitable mechanism for service provisioning for smart infrastructures. The application plane of the SDIC-IoT architecture provides a platform to develop novel applications for customized traffic forwarding based on the requirements of the underlying infrastructures. We discuss the available implementations of the SDIC-IoT architecture in the next section.

IV. TAXONOMY OF SOFTWARE DEFINED INFORMATION-CENTRIC INTERNET OF THINGS

This section discusses the taxonomies of SDIC-IoT using various parameters. Fig. 7 shows the taxonomy of SDIC-IoT approaches. We divide the existing literature into eight categories, with each addressing different aspects of IoT services provisioning.

A. SDIC-IoT Frameworks for IoT

SDN and ICN bring significant innovations in the future IoT service provisioning architecture using various framework implementations. The approaches discussed in this section implement a framework for SDIC-IoT integration for IoT services delivery. Veltri et al. [79] develop the coCONET framework to deploy ICN in the existing SDN architecture. They modify the OpenFlow channel to support coCONET

operations. The coCONET framework is responsible for Interest propagation in the network towards the host which stores the content. The host replies with the content by sending the requested content to the source. The framework develops a sustainable content naming strategy that is used throughout the network for content routing. It allows seamless interactions among users and Interest data items. Wang et al. [80] develop a novel framework to integrate different ICNs on top of the same physical network. They deploy ICN architecture on an Open vSwitch at the data plane. This framework uses content distribution management, forwarding path calculation, name to tag mapping, and OpenFlow strategy dissemination for IoT services delivery. Eum et al. [81] distribute SDICN framework layout into edge networks, the OF-ICN forwarding network, and the OF-ICN control network. The OF-ICN forwarding network transfers data collected through sensors at the edge network. The OF-ICN control network acts as a centralized control authority to optimally route network traffic. Arumaithurai et al. [82] propose an ICN-based service chaining framework that complements SDN characteristics. It supports scalability, flexibility, and reliability in the underlying network. The controller selects an optimal path for traffic forwarding through Open vSwitches, which implement service chaining. However, this technique stacks the data packet labels aggravating the limitations of searching accurate service labels from the stack. Xiulei et al. [83] propose a low complexity SDICN framework based on OpenFlow. This framework ensures optimal data collection from target data centres using predefined rules installed at data plane devices. It also provides load balancing by selecting optimal paths from source to destination. Siracusano et al. [84] deploy an experimental framework of ICN over SDN using virtual and physical testbeds. They develop a reference environment by updating control logic to implement optimal caching policies over the SDN. However, the scope of the proposed caching strategy is limited, and Mininet-based experimental evaluation suffers from simulation errors when deployed on an actual service provisioning environment. Liu et al. [85] simplify the control of space-terrestrial networks using SDN and ICN. They use request aggregation and data distribution of forwarding nodes to control the network and reduce delays flexibly. However, Least Recently Used (LRU) and Least Frequently Used (LFU) caching strategies consume higher network resources. Thus, efficient cache replacement strategies are still required to limit network resource consumption.

The existing SDIC-IoT frameworks support the virtualization of the network resources and deploy ICN characteristics at data plane components. However, most of them don't consider the various resource-limitation constraints of IoT devices. The limited battery power causes data unavailability issues when an underlying device is dead due to battery depletion. Therefore, novel SDIC-IoT frameworks must consider the resource-limitation constraints of IoT.

B. SDIC-IoT for IoT Mobility

Mobility is an essential aspect of IoT due to the mobile nature of content-capturing devices. SDIC-IoT efficiently addresses content mobility issues using global content naming,

Fig. 7. A taxonomy of existing literature based on different aspects of IoT services provisioning using SDIC-IoT.

routing, and caching strategies. Ren et al. [86] propose a mobility management approach for CCN using SDN control logic. The controller ensures mobility by dynamically updating routing tables at the data plane, allowing simultaneous updates of content location and names. However, weak experimental evaluation limits its applicability in a real environment. Zhang et al. [87] propose a QoS-aware SDICN integration for satellite-terrestrial networks. This scheme supports content delivery, mobility, in-network caching, and content-based traffic engineering. However, latency and energy consumption increase during a continuous interaction of the data plane and control plane. Therefore, this approach requires design considerations for a low latency control plane. Xing et al. [88] propose a wide area deployable SDICN approach, which separates edge services network and core transmission network. Edge services network connects resource-limited devices and implements intelligent traffic routing and caching. This scheme supports mobility and a higher transmission rate by retaining traditional IP techniques in the core transmission network. It supports unicast, multicast, mobility, multihoming, and multipath transmission. However, the authors perform small-scale experiments that cannot predict its applicability in a real environment. Further, optimized path selection for the content has not been considered. Ravindran et al. [89] propose mobility as a service over 5G-ICN using application-driven network slices. This technique delivers services over ICN in a virtualized control and data plane to overcome complex network logic. However, communication among different network slices brings scalability issues. He et al. [90] propose a mobility-aware low-complexity resource allocation strategy in SDICN. They develop an integration scheme, which jointly considers networking, caching, and computing for performance enhancement in vehicular networks. This framework uses deep reinforcement learning to devise a resource allocation strategy. However, the integration of heterogeneous

platforms still increases the complexity of the system. Leal et al. [91] propose an integration framework of the SDICN to meet mobility requirements of command and control on the Internet of Battle Things (IoBT). SDN deals with the traditional IP networks, whereas ICN supports heterogeneity of IoBT. However, smooth SDN controller migration in SDN and ICN control still needs novel solutions. Benedetti et al. [73] propose SDICN for mobility support to overcome limitations of lack of state forwarding information in traditional ICN. This technique aims to evaluate the energy consumed by intermediate routers considering different mobility and application settings. However, the authors [73] does not consider optimization strategies in the presence of multiple mobile consumers and SDN controllers. Ban [92] proposes a behavioural analysis strategy using convolutional neural networks on the data collected through SDICN. However, this technique suffers from low convergence speed and unawareness of data size intervals.

Content forwarding strategies using SDIC-IoT must be adaptive enough to deal with the challenges of node failure in the context of large IoT deployments. ICN supports consumer mobility by design; therefore, novel techniques for IoT content producer mobility should be developed. In this case, MobilityFirst provides a dedicated example to deal with this challenge [93].

C. SDIC-IoT OpenFlow Implementations for IoT

These techniques utilize SDN OpenFlow to implement caching, routing, and traffic forwarding strategies. The generated IoT content could be cached based on optimal route selection strategies supported by the controller. OpenFlow then deploys content forwarding rules on the data plane devices and delivers content from producers to consumers. Vahlenkamp et al. [49] deploy ICN over existing ISP and data center networks using SDN. The state is stored in network elements, which reduces the extra time required to contact the controller. Salsano et al. [60] deploy CONET as an extension of southbound OpenFlow protocol. Similar extensions could be developed for northbound API, which offers more possibilities to program SDN. Li et al. [94] deploy ICN over SDN-controlled IP networks. They build a testbed to identify content requests, routing, and delivery. OpenFlow controls the data plane Open vSwitches, which forward ICN content as per Forwarding Information Base (FIB) computed by the controller. They modify the OpenFlow using NFV to implement ICN. This design is based on CCNx (virtual implementation of ICN), floodlight, and Open vSwitch. However, it suffers from scalability issues when multiple content holders request the controller for FIB. Mougy et al. [95] discuss architectural aspects of ICN over SDN by specifically focusing on the standardization of southbound OpenFlow. The authors demonstrate the need for the distributed control plane for the scalability of the SDICN architecture. Gao et al. [96] propose a scalable architecture for intra-domain communication to address controller scalability challenges. This technique uses multiple domain controllers to allocate resources at different levels. However, simulation-based experiments don't depict the actual implementation

paradigm. Petropoulos et al. [97] propose a scalable and dynamic network topology bootstrapping to implement ICN over SDN. This technique extends the OpenFlow to decouple content requests and forwarding phases. It forms a scalable and efficient interoperability framework, which could be deployed over SDICN for IoT service provisioning. Vassilakis et al. [98] propose ICN protocol for SDN-based cellular networks. The authors develop two novel network functions, namely the request resolution function and content routing function. However, this approach suffers from the lack of multi-source content delivery. Kalghoum et al. [99] propose NDN architecture based on SDN for network service provisioning. They devise a routing strategy by decoupling control and data planes. They analyze the switch flow tables and develop a cache replacement policy. Ai et al. [65] propose network recovery by alleviating network congestion using SDICN. The system continuously performs network surveillance, detects discrepancies, and performs recovery of the network. Amadeo et al. [100] propose an architecture of NDN data plane coupled with SDN control plane for independent traffic routing. However, this technique lacks service allocation strategies that are crucial in the optimal delivery of content.

ICN-specific implementations of OpenFlow require name matching for content forwarding at the data plane. OpenFlow should be able to address flows stored at the data plane components through content names. Novel techniques for flow handling should be developed to identify the packet type, content name, and content segment.

D. SDIC-IoT Virtualization Approaches for IoT

In these techniques, NFV supports ICN by virtualizing network elements. NFV enables flexible network virtualization as it offers on-demand network function deployment at general-purpose hardware. This way, commodity hardware could be programmed to offer ICN capabilities of in-network caching and name-based routing. Zhang et al. [101] propose a solution for the challenge of time-sensitive data transmission using ICN virtualization. The user initially requests data from the network controller. The network controller then finds the data source and identifies a data delivery path for efficient data delivery. However, the proposed solution suffers from network congestion issues when multiple nodes request the same content. Li et al. [102] propose an integration of ICN with NFV to support augmented reality-based content sharing. This scheme eliminates the imbalance distribution of content requests and available resources by virtualizing the resources and offering on-demand network slices to the requesting nodes. However, this technique suffers from the lack of optimization strategies. Liao et al. [103] propose a cognitive popularity-based service distribution architecture using SDICN. They develop an SDICN-enabled edge training scheme to generate an accurate AI service model over decentralized big data samples. AI service caching and distribution schemes provide user request-oriented cognitive popularity models for content caching and distribution optimization.

Combining SDN and NFV lead to an agile infrastructure capable of orchestrating IoT services. New NFV techniques

should consider ICN implementation over SDN through a certain amount of programmability of data plane elements. A novel virtualization platform based on the already available implementation of SDN could be explicitly developed, including the ICN layer capable of addressing requirements of IoT.

E. SDIC-IoT Layered Implementations for IoT

The current implementation of OpenFlow might not provide an abstraction for ICN content. In this case, per-flow granularity might not work for content where an operator must assign different forwarding rules to other contents. The techniques discussed in this section use layered extensions in SDN to integrate ICN. Syrivelis et al. [104] propose a layered architecture, which converts the flow concept of SDN into forwarding identifiers to address the content. However, the scalability of this approach depends on the capacity of the OpenFlow channel and its ability to respond to incoming requests. Thus, it suffers from a slower response time at a higher number of content requests. Huang et al. [105] develop a data-driven information layer in the SDN reference architecture to promote network intelligence, configuration management, deployment, and programmability. It consists of four platforms, including the measurement platform, data management platform, data analysis platform, and server platform. A coordinated effort of these platforms provides higher performance and flexibility during content delivery. Saadeh et al. [32] propose a privacy-aware IoT architecture based on the combination of ICN and SDN. This architecture distributes control functionalities over different layers, including the operational, tactical, and strategic layers. It provides more abstraction levels for node interactions in the network, supporting interoperability. However, this architecture does not consider protocol upgrades and implementation. Bracciale et al. [6] propose a lightweight named object solution to represent IoT objects in a derived namespace using ICN. They propose an abstraction layer capable of ICN-based interactions among IoT objects. This technique provides easy discovery, programmability, and management procedures for IoT. This technique defines forwarding agents-specific rules to route traffic securely. Luo et al. [106] propose a layered approach for ICN and SDN integration to reduce flow setup delay, flow setup requests handling, and flow rule installation. They use Color architecture into SDN, devising an SD-Color paradigm for ICN implementation. Zhang et al. [107] propose a layered approach using 5G for optimal in-network caching locations, NFV to derive optimal data content, and SDN to dynamically configure wireless resource allocation strategy. This technique jointly optimizes the effective user capacity and resource allocation fairness and coordinates non-cooperative naming among all users. Tariq et al. [108] propose a layered approach to integrate ICN, SDN, and IoT to devise an energy-efficient priority-based forwarding. This technique uses efficient flow management of the SDN controller to control the broadcast storm and efficiently forward priority-based packets.

An ICN service layer could be implemented on top of the already available layered architecture of SDN. This could lead to a solid implementation of a southbound interface that

TABLE V
AN EVALUATION OF EXISTING SDIC-IoT APPROACHES DIVIDED INTO DIFFERENT CATEGORIES BASED ON DIFFERENT ASPECTS OF IoT SERVICES PROVISIONING.

Research	Category	Scalability	ICN Use Case	Mobility	Application Domain	Virtualization	Solution Architecture	Features
Veltri et al. [79]	SDIC-IoT frameworks for IoT	Low	CONET	×	IP networks	Medium	Architecture	OpenFlow modification
Wang et al. [80]		Low	General	Low	IP networks	Low	Architecture	Open vSwitch
Eum et al. [81]		Low	General	Low	IP networks	Medium	Architecture	Data dissemination
Arumathurai et al. [82]		High	General	Low	IP networks	Medium	Architecture	Service chaining
Xiulei et al. [83]		High	CCN	High	IP networks	High	Solution	OpenFlow modification
Siracusano et al. [84]		High	CONET/CCN	High	IP networks	High	Solution	Deployment of SDICN
Liu et al. [85]	SDIC-IoT for IoT mobility	Low	ICN	Medium	Space networks	Medium	Architecture	Low latency
Ren et al. [86]		High	CCN	High	Future networks	High	Solution	SD Controller
Zhang et al. [87]		Low	ICN	Low	Terrestrial networks	Low	Architecture	QoS
Xing et al. [88]		Medium	ICN	Medium	General networks	High	Solution	Intelligence
Ravindran et al. [89]		Low	5G-ICN	Low	5G networks	Low	Architecture	Network slicing
He et al. [90]		Low	ICN	Low	Vehicular networks	Low	Architecture	Joint optimization
Leal et al. [91]		Low	ICN	Low	IoBT	Low	Architecture	Command control
Yunqiang Ban [92]		Low	NDN	High	Business networks	Medium	Architecture	Energy efficiency
Benedetti et al. [73]		Low	NDN	High	IP networks	Low	Architecture	Consumer mobility
Vahlenkamp et al. [49]		Low	CCNx	Low	IP networks	Low	Architecture	ICN routing
Salsano et al. [60]		High	CONET	Low	IP networks	High	Solution	Northbound API
Li et al. [94]		Low	CCNx	Low	IP networks	Low	Architecture	Open vSwitch update
Mougy et al. [95]	SDIC-IoT OpenFlow implementations for IoT	-	ICN	-	Intra-domain	-	-	Survey
Gao et al. [96]		High	ICN	Medium	Intra-domain	High	Solution	Distributed controllers
Petropoulos et al. [97]		High	PURSUIT	Low	ISP	Low	Prototype	Interoperability
Vassilakis et al. [98]		High	ICN	Low	Cellular networks	Medium	Prototype	Request resolution
Kalghoum et al. [99]		Low	NDN	Low	IP networks	Low	Solution	Cache replacement
Ai et al. [65]		High	ICN	High	IP networks	High	Solution	Fault-tolerance
Amadeo et al. [100]		Low	NDN	Low	IP networks	Low	Architecture	Fog computing
Zhang et al. [101]		High	ICN	High	Cellular networks	High	Solution	Delay-bounded QoS
Li et al. [102]		High	ICN	High	5G networks	High	Solution	AI services
Liao et al. [103]		High	ICN	High	Vehicular networks	High	Solution	Caching localization
Syrivelis et al. [104]	SDIC-IoT layered implementations for IoT	Low	ICN	Low	IP networks	Low	Prototype	Forwarding functions
Huang et al. [105]		Low	ICN	Low	IP networks	Low	Architecture	Content management layer
Saadeh et al. [32]		Low	ICN	Low	IP networks	Low	Architecture	Cloud support
Bracciale et al. [6]		High	NDN	High	IP networks	-	Solution	Named object solution
Luo et al. [106]		Low	NDN	Medium	IP networks	Low	Architecture	Flow setup efficiency
Zhang et al. [107]		High	NDN	Medium	5G Wireless networks	High	Solution	QoS enhancement
Tariq et al. [108]		High	NDN	High	Smart grid	High	Solution	Priority packets
Liu et al. [109]	SDIC-IoT optimal caching node selection for IoT	High	ICN	High	IP networks	High	Solution	Deep learning
Liu2017Internet et al. [110]		Low	ICN	Low	IP networks	Low	Architecture	Integer programming
Turjman et al. [111]		High	ICN	Medium	IP networks	High	Solution	Fog caching
Nascimento et al. [112]		Medium	ICN	Medium	IP networks	Medium	Solution	Bandwidth optimization
Xu et al. [113]		High	ICN	High	IP networks	High	Solution	AI-based caching
Zhang et al. [114]		High	ICN	High	IP networks	High	Solution	Integer linear programming
Chanda et al. [115]	SDIC-IoT traffic management for IoT	Medium	CCN	Medium	IP networks	Medium	Architecture	Resource optimization
Chen et al. [116]		Medium	MobilityFirst	High	IP networks	-	Architecture	Service-oriented
Lv et al. [117]		High	ICN	Medium	IP networks	High	Solution	Community routing
Cai et al. [118]		Medium	CCN	Medium	IP networks	Medium	Architecture	Status aware routing
Zhang et al. [20]		High	CCN	Medium	IP networks	High	Solution	Community division
Li et al. [72]		Medium	ICN	Medium	Terrestrial networks	Medium	Architecture	Content retrieval

provides security against data plane tampering and spoofing attacks. ICN layer extracts content metadata at the network layer without DPI. The content-centric nature of ICN supports data processing at the network layer. Content metadata contains name, length, and type, making it flexible for network operators to optimize network behaviours based on content.

F. SIC-IoT Optimal Caching Node Selection for IoT

Caching techniques support instant information access, storage, and efficient utilization of network resources. Conventional cache replacement systems send multiple client requests through the same connection to obtain the same content. However, SDN-based techniques optimize content caching strategies using centralized control of the network infrastructure. Liu

et al. [109] propose a deep learning-based content popularity prediction based on the computing resources of switches. This technique improves caching performance by predicting content popularity. Liu et al. [110] propose cache placement for popular high resource-intensive content. The system design includes a centralized control architecture duly equipped with a cache controller. Cache placement is formulated as an integer programming problem. An integer-based knapsack strategy solves this problem and finds optimal cache placement nodes. Turjman [111] propose a cache replacement strategy based on the periodic request popularity of on-demand requests and IoT active mode to transmit the captured content. A number is assigned to cached data based on a cached timestamp to retain the most valuable information in the cache. Nascimento

Fig. 8. The controller manages data plane devices and implements caching on selected nodes.

et al. [112] propose a cache replacement strategy using SDN to avoid duplicate requests in the same connection. They use the least connection load balancing algorithm to optimize resource consumption in the network. Xu et al. [113] propose a joint resource scheduling strategy to manage the compute and caching resources uniformly. Resource allocation provides short-term and long-term caching strategies. The short-term strategy caches the content on intermediate nodes, whereas the long-term caches the AI model. Zhang et al. [114] propose a caching strategy for resource-intensive network traffic. They transform the resource allocation problem into a dynamic integer linear programming problem. They optimize caching strategy based on video popularity and transmission delay.

Fig. 8 shows SDN-based caching strategy. The controller selects the optimal path for caching node selection. Thus, achieving efficient content placement and retrieval. Although caching strategies enhance content delivery, indiscriminate caching may cause content replication and localize caching decisions. Novel techniques need to be developed using SDN to identify cache status, thereby avoiding content replications optimally. SDN control plane can implement on-path/off-path and hybrid approaches to avoid content duplication. Another drawback of existing techniques is that content providers are unaware of the content they provide due to the publish/subscribe model. This raises more content duplication issues.

G. SDIC-IoT Traffic Management for IoT

ICN characteristics of in-network caching and name-based routing significantly change routing and transmission of Internet traffic. Therefore, traffic engineering techniques need to be revisited. Data plane routers can collect content metadata, which makes it easier to keep track of the data volume transferred on each link. Chanda et al. [115] propose a content metadata access-based traffic engineering scheme. They

observe content metadata at the network layer to optimize traffic routing among different nodes. The optimization criteria could be different based on network operator priorities such as caching and bandwidth utilization. Chen et al. [116] propose a traffic engineering technique to support service provisioning in IoT. They use global name resolution and globally unique identifiers of MobilityFirst to represent IoT services. Lv et al. [117] propose a traffic routing technique by dividing ICN topology into different communities based on maximal tree. The information is kept in the relevant information centre community, where information forwarding is performed using intracommunity and intercommunity schemes. Intracommunity traffic scheme forwards network traffic based on the same community while intercommunity strategy routes network traffic based on social relationships. Cai et al. [118] analyze traffic forwarding priorities based on packet delay and node load. The controller updates forwarding priorities of the matching interfaces according to the network status. The highest priority Interest packet is delivered through the matching interface. Zhang et al. [20] propose an intelligent routing mechanism based on SDN and community division. This technique uses eigenvector centrality and label propagation for content retrieval. A controller is then deployed for each divided community, which efficiently routes network traffic. Ant Colony Optimization moderates intracommunity communication, and intercommunity routing is based on the frequency of interactions. Li et al. [72] propose SDICN to cope with rapid access requests and management of heterogeneous networks. They develop a three-layer architecture. A novel on-board switch based on Protocol-Oblivious Forwarding (POF) is designed, which performs cooperative and coded caching.

Traffic engineering techniques deal with traffic burstiness and improve resource utilization by finding optimal routes to reduce content delivery. Name-based data packets can be viewed as flows; this transforms the definition of flows in IP networks. Therefore, fine-grained flow management techniques are required in SDIC-IoT. Since the number of content names in ICN is enormous, cache placement and traffic engineering are NP-hard problems. Therefore, novel techniques need to be proposed to solve traffic engineering and cache placement problem.

H. Summary and Insights

Table V shows the taxonomy of existing SDIC-IoT schemes based on different features. Any IoT service provisioning solution should consider scalability, mobility, and virtualization. Therefore, we divide the level of these characteristics into low, medium, and high among existing approaches. QoS support and consistency of the content delivery in SDIC-IoT need new solutions. Efficient middleware techniques are required that ensure expressiveness while maintaining information forwarding at a feasible rate. The relationship between resource-awareness and resource adaptation needs to be explored in future studies. The programmability characteristics of the controller can be used to control network resources intelligently. Other technologies such as big data, cloud computing, and wireless access networks can be integrated to make SDIC-IoT

more secure and reliable. One of the critical characteristics of ICN is the ability to cache information at intermediate nodes and use it collaboratively for information sharing throughout the network. Optimizing caching through cloud replication brings benefits for efficient bandwidth utilization and reduces transmission delays.

V. USE CASES OF SOFTWARE-DEFINED INFORMATION CENTRIC INTERNET OF THINGS

In this section, we present an overview of recently developed synergies and case studies of SDIC-IoT. A summary of these case studies is shown in Table VI.

A. Smart City

Smart cities use interconnected objects for ubiquitous service provisioning. IoT improves citizens' quality of life by supporting smart city systems, including smart power control, smart parking systems, smart building management, and smart security services. Smart cities use a massive number of IoT objects which require a large namespace, interoperability, and security services. Many integrated systems provide applications and services in smart cities, including homes, buildings, vehicles, hospitals, and public infrastructures. IoT-based smart cities enhance different aspects of the human lifestyle. For example, finding suitable parking slots becomes challenging due to the growing number of vehicles and the lack of parking spaces. Intelligent parking management systems use smart sensors to identify available parking spaces and related costs to automate the management of parking services. Citizens can subscribe to parking services; they are notified when specific parking slots are available. In this case, SDIC-IoT provides a seamless communication model where the subscribers are interested only in the information they need, e.g., parking slots. Another example of using distributed sensors is traffic congestion management at specific crowded locations in the city. Citizens may get real-time warnings about traffic congestion levels, which limits the chances of catastrophic incidents. Deep learning could be integrated with SDIC-IoT to proactively notify citizens at workplaces, schools, and public places.

He et al. [119] propose an SDN-based solution for smart cities where content caching is enforced on edge cloudlets. Deep reinforcement learning identifies resource caching policy considering content location and consumption. The SDN controller manages the underlying network equipped with content caching and computing capacities. A cache hit occurs when a user wants to access the data, or smart city applications need to process the data. Due to a large number of network states, deep learning can be used to identify which particular virtualized resources are assigned to a specific user. A significant limitation of this system is the consumption of a vast amount of energy when implementing deep learning models on various edge servers in the network. Moreover, long naming convention produces higher overhead in the network. Fekih et al. [120] propose a comparative study of using CCN and SDN for smart cities. They utilize a modified OpenFlow to envision ICN functionalities. They divide the network into

sub-networks where distributed controllers manage the sub-nets. A root controller gathers information from distributed controllers to optimize resource utilization and cache management. A major limitation of this system is the lack of actual implementation. Although the theoretical model provides an insightful understanding of the proposed solution, the lack of experimental evaluation brings trust issues. Liao et al. [103] propose a cognitive popularity-based intelligent service delivery framework. They design an SDICN-enabled training scheme implemented on the edge cloud. The training scheme uses decentralized big data samples collected throughout the network to generate accurate models. The caching scheme automatically identifies content popularity based on the number of content requests. Although the scheme shows promising results for AI caching and distribution, training of big data models produces huge network overhead causing delays in content caching and distribution.

Due to a lack of comprehensive solutions dealing with end-to-end service delivery, integrating SDN and ICN requires further research efforts. Flexible integration models are required to address data dissemination issues among heterogeneous infrastructures. Furthermore, there is a need to develop scalable caching solutions where content could be cached at a city-wide scale.

B. Smart Homes

Smart home services aim at improving personal lifestyle by capturing real-world user information supported by IoT. These services include home security, temperature monitoring, fire detection, and lighting control. These services support remote home appliance control by collecting environmental data and sending real-time conditions to the controllers. Novel home control solutions provide full control of the home automation, i.e., autonomous light control, home security using face recognition, and automatic temperature control based on outdoor conditions. Smart home automation doesn't require the address of the hosted sensors or services; instead needs information about the essential services. Access control needs an authorization mechanism based on the content itself rather than the location of the sensor. SDN helps heterogeneous device management in smart homes. The data produced by sensors for home management could be cached at optimal locations identified by the SDN controller. The network control traffic should be adaptable enough to coordinate with the conventional control schemes in this paradigm. Therefore, SDIC-IoT as a communication model helps faster deployment and management of services in smart homes.

The authors in [121] compare IPv6 proposal with ICN in smart homes using control, service, and complexity parameters. The authors discuss heterogeneity challenges in ICN and IoT integration. They use ICN features of naming and in-network caching for content dissemination and mobility handling. The authors propose a service discovery protocol and content-based policy for user interactions with IoT devices. Amadeo et al. [59] propose an ICN framework for smart homes containing a flexible and expressive naming scheme, which supports information exchange and configuration management. The proposed framework applies to common smart

TABLE VI
AN OVERVIEW OF RECENTLY REPORTED SYNERGIES AND CASE STUDIES OF SDIC-IoT.

Use Case	Research	ICN Variant	Scope	Limitation
Smart cities	He et al. [119]	Dynamic orchestration of networking caching, and computing resources.	ICN on edge cloudlets based on deep reinforcement learning.	Energy Consumption
	Fekih et al. [120]	Modification of OpenFlow, root controller, cache placement.	CCN and SDN for smart cities.	lack of experimental evaluation brings trust issues
	Liao et al. [103]	Content popularity.	Cognitive popularity-based AI service distribution.	Delay in content caching and distribution.
Smart homes	Naeem et al. [121]	NDN, Naming and in-network caching, for content dissemination and mobility handling.	Control, service, and complexity, ICN and IoT integration that arise due to IoT heterogeneity.	Communication overhead.
	Amadeo et al. [59]	naming, caching, forwarding, retransmissions, and security.	Flexible and expressive naming scheme supporting information exchange and configuration management.	Scalability issues.
	Burke et al. [122]	CCN, Naming, security, access control.	SDN-based security using trust models with key attributes	Limited access control scheme.
Smart Healthcare Systems	Halil et al. [123]	NDN service caching, naming structure.	Abstract caching for heterogeneous IoT.	Limited mobility support.
	Nour et al. [57]	NDN architecture for traffic models.	Mobility using mapping-based and hand-off mechanisms.	Lack of security.
	Zhang et al. [124]	Data access using NDN.	Hierarchical naming strategy and security of data.	Communication overhead.
Smart Grid	Han et al. [125]	Communication, on-path caching routes.	Layered approach, publish/subscribe model to organize communication independent.	Lack of implementation.
	Katsaros et al. [126]	ICN solutions.	Multi- rendezvous point selection and in-network processing.	Lack of resource optimization.
	Zhang et al. [123]	ICN-based framework for the home energy management.	Aggregation and analysis for informed decision making.	Scalability issues.
Smart Transportation Systems	Li et al. [102]	NFV and ICN-based service orchestration.	Content sharing scheme, imbalance among the content request and the resource limitation.	Increased reconfiguration cost.
	Bouk et al. [127]	NDN and smart transportation systems.	Secure and reliable communication.	Added complexity.
	Amadeo et al. [3]	ICN caching, naming, and mobility.	ICN characteristics for transportation challenges.	Increased latency.

home service models, including push, pull, and multiparty. Nodes in the network implement NDN characteristics developed explicitly for data and management planes of the network. The Data plane controls the data and Interest packets and ensures naming, caching, request, retransmissions, and security. The management plane configures, control, and manages network functions on the data plane elements. The authors in [122] identify security requirements in home automation and develop an SDN-based security framework. They integrate the trust with key traffic features and implement access control using the name of the content. Thus, services are identified and authorized using content names. The sensor analyzes the access control signature, processes it, and replies with a data packet in response.

Although the existing literature focuses on SDIC-IoT in smart homes, more research is required to develop large-scale solutions. Users of unsolicited content interrupt Interest configuration as persistent Interest is not always enough to handle an increasing number of content subscribers. Moreover, secure content dissemination is necessary to enable authorized subscribers to consume the content. Privacy of user information is also highly essential as smart homes deal with highly personal user information.

C. Smart Healthcare Systems

Smart healthcare is one of IoT's most prominent use cases due to its implications in remote patient monitoring, patient tracking, disease diagnosis, augmented surgeries, and emergency care. Smart healthcare is a coordinated strategy involving sensors and actuators, wearables, smart pills, wireless networks, and mobile applications. Smart sensors capture content from physical devices, in-body sensors, and wearables. Personal mobile devices are also used to monitor and track patient physical conditions; this provides insights for

appropriate treatments. Environmental sensors measure atmospheric conditions around patient vicinity, including humidity, temperature, and air quality. Different healthcare sensors leverage communication infrastructure to facilitate ubiquitous healthcare services. Healthcare professionals monitor real-time patient conditions using data collected by the sensors. The patient record is maintained continuously, which enables future diagnosis and treatments. Third-party applications such as insurance companies can view patient health status based on permission levels. Patient information can be cached for instant access and availability at any network node, including home gateways, hospital databases, and clinical access points. Patient information can be shared among different entities through patient names, which eases information acquisition and sharing. Furthermore, security and access control rules can be implemented on the user's name, making it easier to access as compared to using the address of the host node. Moreover, OpenFlow of SDN could be combined with ICN to optimize content caching in smart healthcare systems.

Halil et al. [123] propose an IoT-based architecture for healthcare service caching over NDN. They develop a special naming structure to solve heterogeneity issues and integrate different services among heterogeneous infrastructure. However, a significant limitation of this study is that it provides an abstract implementation of the caching scenarios. However, it doesn't consider the secure mobility of patients for seamless publish/subscribe content dissemination. Nour et al. [57] propose NDN for healthcare services for different traffic models supporting service management. This approach supports producer and cache mobility using mapping-based and hand-off mechanisms. However, network content can always be accessed by any node, thereby requiring the security of the communication channel. Zhang et al. [124] propose an NDN-based distributed mobile health architecture. It supports secure

content distribution rather than the expensive communication medium. This architecture promotes direct content access using a cloud-based mobile platform. They use a hierarchical naming strategy independent of the content producer; it implements security on the content, making it self-secure. Saxena et al. [128] propose an NDN-M2M communication strategy for smart healthcare. It captures and transmits content produced by sensors to the local cloud server and identifies emergency events using Hidden Markov Model. An adaptive content-aware forwarding scheme sends events to the home cloud, and exchanges the required content using publish/subscribe strategy.

Healthcare uses different content access levels based on the sensitivity of healthcare data. Therefore, novel access control techniques should be developed with name-based access control for different stakeholders. Emergency situations require specific QoS parameters for patient treatment; thus, event tags should be generated to distinguish various medical conditions. Furthermore, reliable content delivery should be ensured from mobility-prone medical equipment.

D. Smart Grid Systems

The smart grid is an electricity supply network, which uses computing infrastructure to optimize power management [129]. Smart grid systems aim at improving power consumption in homes, buildings, and industrial control systems. They collect, analyze, control, and manage energy transfer networks to ensure fault-tolerance, service delivery, and improved QoS. IoT in smart grids supports customer-side energy management by providing intelligent power consumption, energy management, interoperability, and charging/discharging equipment. The smart grid enables efficient energy management by supporting backup and recovery solutions. They support power distribution during peak hours and help reduce electricity costs. ICN supports seamless communication in a smart grid where any node can fulfil content requests. Many smart grid systems use M2M communication, where communication patterns are often predictable. ICN can support targeted in-network caching for efficient M2M communication. Machine-critical smart grid systems suffer from network intrusions resulting in severe economic loss. ICN security solutions particularly help in this paradigm since host locations are not exposed. Moreover, the content naming scheme supports optimized multi-source content delivery, which reduces traffic congestion.

Han et al. [125] propose on-path time synchronization for smart grids using SDIC-IoT. The SDN controller computes on-path caching routes before choosing the nearest nodes to implement accurate time synchronization. They use the publish/subscribe model to organize communication independent of location. The proposed system consists of three planes: OpenFlow ICN control network, OpenFlow ICN network, and edge network. OpenFlow ICN control network replicates control plane whereas OpenFlow ICN plane corresponds to forwarding plane in traditional SDN. Edge networks cover smart grid entities capturing physical world parameters. Katsaros et al. [126] employ ICN in smart grids to solve

various communication challenges, including fault management, energy management, and resource optimization. They compare the performance of ICN and host-centric solutions and report that ICN performs better in emergencies using multi-rendezvous point selection and in-network processing. Zhang et al. [123] propose an ICN-based framework for home energy management systems. This system collects data from different household elements in smart homes and performs aggregation and analysis for informed decision-making. The architecture of this framework uses two layers, including the publish/subscribe layer and the security layer. The security layer is responsible for sharing security keys among subscribers. However, a significant drawback of this technique is that it does not scale with the increase in data. More investigations are needed to define M2M communication on top of content names. Moreover, secure publish/subscribe content flow between active subscribers requires flexible key exchange protocols.

E. Smart Transportation Systems

Smart transportation uses communication and computation infrastructure to manage transport networks [130]. IoT plays a key role in this context, including traffic control systems to monitor network traffic, network services deployment to route traffic, and sensors to provide autonomous parking services. These services improve the safety, security, accuracy, and performance of the transportation infrastructure. Public transportation significantly impacts human lives by sharing large volumes of data during Vehicle-to-Vehicle (V2V) and Vehicle-to-Infrastructure (V2I) communication. Any discrepancy during communication may endanger human lives due to the critical nature of smart transportation systems. Efficient data exchange techniques should be developed to avoid network congestion, thereby promoting the safety of the citizens. Thus, caching schemes using ICN help in data availability near vehicle locations. Another scenario where SDIC-IoT could be applicable is when users subscribe to different transportation services to identify vehicle routes, delays, and congestion. These services may combine both real-time and legacy information for personal needs and fulfilment of the service demands. Transportation services could be delivered based on the name of the content regardless of the location where these services are originated. SDN in this paradigm deals with heterogeneous devices using OpenFlow. It hides the complexities of the mobile nodes connected using different infrastructure elements and communication protocols.

Li et al. [102] propose optimal resource allocation and service management strategy for vehicular networks using NFV and ICN. They develop a content-sharing scheme that eliminates the imbalance between content requests and demand by allocating on-demand virtual resources. Bouk et al. [127] propose a novel architecture for integrating NDN in smart transportation systems for mobility-aware security. They implement a secure and reliable communication mechanism among mobile devices using NDN characteristics of in-network caching and content-based security. Amadeo et al. [3] propose a framework for integrating ICN and vehicular

networks. They use ICN characteristics of naming, caching, and multicasting to overcome smart transportation challenges. Saxena et al. [131] discuss NDN-based content routing techniques in vehicular networks for content dissemination by implementing different IP-based forwarding schemes.

Apart from the existing literature, mobility is one of the prime characteristics of intelligent transportation systems, which influences QoS. Collaborative caching schemes can be developed by integrating edge computing to provide content in a mobility scenario. Moreover, adequate traffic forwarding schemes are required for vehicles using centralized management characteristics of SDN.

F. Lessons Learned: Summary and Insights

Content produced by resource-intensive and delay-sensitive IoT services could be executed on edge cloudlets in the vicinity of IoT. ICN adaptation could provide benefits of implementing caching strategies on edge nodes. Collaborative caching schemes optimize resource placement, such as edge nodes storing actual content whereas routers storing data indexes. After that, optimization techniques could be developed to create cache indexes considering content popularity and cache-hit ratio. Furthermore, ICN inherently supports high-speed networks such as 5G and 6G using mobility, multi-homing, and content security. By using these characteristics of ICN, content could be directly addressed at the network layer. Caching nodes implement content access policies moderated by authentication servers that are prone to increased latencies and outage issues in case of server failures. Efficient access control techniques could allow authenticated users to access cached content without authentication by online access providers or cache routers. However, such policies may suffer from scalability issues due to a large number of authentication requests. Therefore, an optimized and secure caching mechanism is required for smart service provisioning in different SDIC-IoT use cases.

VI. CURRENT CHALLENGES AND FUTURE RESEARCH DIRECTIONS

SDIC-IoT promises content-centric service delivery of modern IoT applications. Although a significant amount of research is available in SDIC-IoT, real-world implementations are non-existent. The communication aspects discussed in the existing literature should be implemented in the core design. The problem with add-on protocols is that they produce extra overhead during communication. It is critical to identify challenges associated with different SDICN components from an IoT perspective. We discuss various implementation challenges of SDIC-IoT and highlight guidelines and future research directions. Table VII summarizes these research directions.

A. Content Naming

One of the significant challenges of SDIC-IoT is the number of content items that are expected to grow by order of magnitude greater than the number of hosts. Therefore, the execution time increases for name-based routing between each node in

the source to destination. This increases transmission delays and decreases throughput. CONET provides a solution for this issue by confining routing to a subset of nodes called NRS nodes; this is realized by separating control and forwarding traffic [44]. NDN utilizes a hierarchical naming strategy, which gives flexibility to developers to choose a naming model for applications and services. Naming sensors, physical objects, and IoT devices are necessary for monitoring and management. Service naming is also essential to distinguish services running on the same devices. Content produced by IoT is often smaller in size as compared to the content name. Thus, better optimization solutions are required to handle naming size issues. Assigning version names to different content versions is efficient in large-scale networks as the same data could be produced multiple times. Routing tables will have various entries having similar content with other names. Saving numerous copies of the same content could create storage, processing, and communication overhead. Efficient schemes to bind names with the content are required to avoid content replication. Moreover, ML techniques could be developed to analyze the content and identify the duplicate content.

It is essential that the northbound API is open and standardized to support the portability and interoperability of services. A critical challenge is to populate the vendor-specific list of functions that optimize content caching. The standardization efforts for the northbound API foster the SDIC-IoT idea of bringing innovation in a software-defined way rather than making changes in the hardware components. It can be helpful to define commands to capture statistics on the requests for content from users to populate caching nodes in advance.

B. IoT Mobility Management in SDICN

IoT encompasses mobile devices, including vehicles, health sensors, and ubiquitous devices attached to the human body for stable and reliable service delivery. The mobility of IoT devices poses significant challenges considering different networks. The default design of SDIC-IoT enhances mobility by re-issuing incomplete requests. SDICN supports IoT mobility so that IoT doesn't need to acquire a new IP address when the network changes. However, the concept of mobility is different from both the producer and consumer perspectives. From the standpoint of producer mobility, there is a need to design scalable name resolution schemes with an efficient mechanism to update the entire system—this way, the new name of the content could be synchronized throughout the network. Moreover, the movement of caching nodes poses significant challenges where probabilistic and optimization caching techniques are required to allocate the nearest caching nodes to consumers. Upon receiving an Interest request, new FIB entries are established on ICN caching nodes between content producers and consumers. Any changes in topology or mobility can invalidate FIB entries at the data plane ICN devices. A novel mechanism is required to remove invalid FIB entries on each node using data plane feedback of SDIC-IoT. Novel ML techniques could be used to select candidate caching nodes. From the consumer perspective of mobility, a promising solution is to resend Interests. However, this is not

TABLE VII
A TAXONOMY OF CURRENT CHALLENGES IN SDIC-IoT IMPLEMENTATION AND FUTURE RESEARCH DIRECTIONS.

Challenges	Causes	Solutions
Content Naming	<ul style="list-style-type: none"> • Greater number of content names. • Increased execution time among nodes. • Increased delay and reduced throughput. • Smaller data size as compared to name. 	<ul style="list-style-type: none"> • Routing on specific nodes using CONET. • Optimized solutions for content naming. • Version naming of content. • Schemes for name binding with content.
IoT mobility management in SDICN	<ul style="list-style-type: none"> • Address resolution in different networks. • Lack of producer mobility due to scalability. • Movement of caching nodes. • Topology change impact on FIB entries. 	<ul style="list-style-type: none"> • Re-routing incomplete requests. • Scalable name resolution at network convergence. • Probabilistic caching to select nearest caching nodes. • Replacing invalid FIB entries using data plane feedback.
QoS Support	<ul style="list-style-type: none"> • Large-amount of on-demand services. • Lack of IoT-specific QoS requirements in current networks. • Best effort approach of NDN. • Lack of virtualization in current networks. 	<ul style="list-style-type: none"> • IoT-specific QoS considerations. • SDN-based solutions for cache placement. • Prioritizing QoS parameters. • Routing traffic based on QoS parameters.
Scalability of SDIC-IoT.	<ul style="list-style-type: none"> • Controller overload. • Higher flow setup latency. • Inefficient naming strategies. • Centralized control of the network. 	<ul style="list-style-type: none"> • Group table update. • Distributed controllers. • Hybrid flow rule installation. • Hierarchical naming strategies.
Big Data Challenges for SDIC-IoT	<ul style="list-style-type: none"> • Large amounts of data. • Large amount of data provoke increase computational complexity. • Data delivery in heterogeneous networks. • QoS management in big data. 	<ul style="list-style-type: none"> • In-network processing of data using ICN • Analytics techniques for big data. • Network slicing using NFV. • Optimal delivery path selection.
Caching location prediction using deep learning	<ul style="list-style-type: none"> • Lack of content popularity prediction. • Sequential network content access. • Increased delay and poor caching. • Individual node-based content popularity prediction. 	<ul style="list-style-type: none"> • Intelligent content caching strategies. • Centralized control using SDN. • Deep learning based content popularity prediction. • Caching of neurons on different nodes using SDN.
SDIC-IoT caching for mobile networks Using 5G	<ul style="list-style-type: none"> • Delay-sensitive resource-intensive tasks. • Wireless-channel fading, outside inferences, and mobility. • Statistical and delay bounded QoS. • Utilization efficiency of the spectrum. 	<ul style="list-style-type: none"> • SDN ICN and NFV for optimize the performance. • Caching content near the users. • Interest finding using ML techniques. • Use of SDN for statistical QoS requirements.
Security and privacy	<ul style="list-style-type: none"> • Heterogeneity of IoT applications. • Limited resources. • Lack of device-specific implementations. • Lack of secure communication medium. 	<ul style="list-style-type: none"> • Security applications development at the data plane. • ML-based security solutions using statistical collection. • Secure update patching of security applications. • Device-specific security applications development.

an optimal solution as it produces massive network overhead due to Interest looping. Therefore, novel mapping techniques require lower communication overhead and lower inter-node communication delay to support streamed and subscribed data.

C. QoS Support

QoS helps to monitor the functionality of the system and the resources required to deliver IoT services. It makes service providers accountable for the reliability and availability of the provided services. QoS for IoT services includes power consumption, bandwidth utilization, and mobility. A large amount of network traffic and demand for future smart IoT solutions (e.g., industrial control, healthcare, and smart city) instigate the need for QoS enhancement. However, current SDIC-IoT systems don't consider the requirements of low latency, higher delivery, and low loss rate. Although NDN supports nearby access, it works based on best-effort services. SDN supports optimal caching locations based on the controller ability to identify the overall network topology. Specifically, the controller implements a connection-based virtual circuit service model. This way, the controller plans a route based on the QoS parameters of the services. A comparative analysis of QoS parameters for specific IoT services must be performed, and then a policy should be defined to establish minimum QoS requirements.

D. Scalability of SDIC-IoT

Scalability is one of the prime concerns in SDN-based systems. ICN also suffers from scalability issues due to a large number of content names and caching strategies. The content present in a network is far greater than the number of devices; this causes a state space explosion. Forwarding tables at the data plane have limited TCAM memory, which

may overflow when defining data paths using content names. Heterogeneous IoT services require controllers to store more parameters such as cache status awareness and updates. This may overwhelm the controller and cause network breakdown. Therefore, a backup controller may be deployed to take control in case of the failure of the active controller. Moreover, flow setup latency requires a significant time, causing delays in traffic delivery. Group table mechanisms of OpenFlow version 1.1 could be a handful for flow rule installation for multiple packets. It assigns multiple group table entries to the same group identifier. In this way, the controller can update multiple flows by updating the group identifier instead of updating numerous flows in the data plane. Moreover, multiple controllers could be installed to control the same network serving different controller roles such as master, equal, and slave. Proactively analyzing flows in the network and devising a hybrid flow rule installation strategy can also help reduce scalability issues. In this regard, some flow rules can be installed based on network analysis before the arrival of the network traffic. Furthermore, the choice of naming schema is an intrinsic and fundamental aspect that can impact ICN's scalability. Hierarchical naming strategies could be defined to solve this challenge, providing rich semantics leading to smaller routing table sizes and reduced storage and computation requirements.

E. Big Data Challenges for SDIC-IoT

IoT produces large amounts of data converging to big data becoming highly complex to manage in traditional networks. Big data management in traditional networks has become a widely researched topic for academia and industry. Due to the substantial volume of data, computing complexity, and rapid processing times, big data applications have become challenging to implement efficiently without the support of

the underlying network. The data intrinsic nature of ICN supports large amounts of data processing at the network layer. Traditional ICN could be extended to process in-network data. SDN capability of centralized control identifies, monitors, processes, and manages the state of ICN in-network processing. Moreover, it can help abstract the storage and network capability from user-data interactions and extract meaningful insights using analytic methods. Big data analytics can help guide decisions regarding future network management. For example, SDN-based big data analytics proactively compute traffic routes and configure forwarding rules for specific traffic. Moreover, time-sensitive and bandwidth-intensive time-varying big data services pose more challenges in constrained wireless networks. In this case, network slicing using NFV can offer dynamic reconfiguration of wireless resources. This might become a promising solution to find optimal content delivery paths.

F. Caching Location Prediction Using Deep Learning

In SDIC-IoT, predicting content popularity plays a key role in establishing efficient content-caching strategies. Some previous techniques [132]–[134] propose that all nodes know the content popularity in the network. However, this is not always true resulting in increased delay and poor caching management. Caching efficiency improvement based on the behavior of individual nodes may bring inaccurate results as network information might be influenced by noise and other discrepancies. The SDN central management capability can be used to observe traffic passing through different nodes as the requested content shows spatial locality. Proximity information can be used to identify popular content and to develop content caching strategies. Network content access is sequential where the content passes through different nodes at a specific time. Thus, network content analysis can provide significant information regarding traffic flow. A deep learning network based on SDN could be developed to predict content popularity at different nodes to design content caching strategies. SDN in this situation can be used to develop multi-scale traffic prediction with higher accuracy. SDN switches at the data plane store different traffic features including spatial, temporal, and social correlation. The deep learning model can understand traffic features from a global network perspective. Thus, distributed patterns of content popularity can be extracted due to the model unsupervised learning strategy, where slight variations in the network behavior don't impact the performance of the network. Deep learning neurons could be easily changed on different nodes in the network using SDN characteristics. This could provide the flexibility to deploy, modify, and update deep learning models in the network to predict content popularity.

G. SDIC-IoT Caching for Mobile Networks Using 5G

In-network caching becomes a critical factor in Fifth Generation (5G) networks because high-speed network content could be cached to optimize resource consumption. Supporting multimedia data on the 5G wireless networks brings novel

Fig. 9. Security and privacy attacks on SDIC-IoT.

challenges that were not present in traditional networks. Multimedia data transmission over 5G wireless networks is both delay-sensitive and resource-intensive, requiring optimized data delivery. 5G wireless channels are more time-varying due to the wireless-channel fading, outside inferences, and mobility. This involves novel statistical and delay bounded QoS ensuring mechanisms. Although 5G wireless infrastructure contains limited resources (e.g., RF-spectrum, battery power, and processing power), the utilization efficiency of the spectrum must be optimized. Moreover, 5G wireless networks consist of heterogeneous wireless and wire-line networks and diverse interfacing architectures. End devices connecting to these networks require cost-effective control mechanisms to optimize their overall operating performance. To address these challenges, SDN, ICN and NFV can be integrated to optimize the performance of 5G wireless networks. The frequently requested content can be cached near to the user groups who have social connections. Common Interest could be identified using ML techniques. NFV techniques can divide the underlying network into multiple virtual networks to execute various applications at different virtual networks. These virtual networks may use the high-speed 5G networks to enable parallel task execution of various applications. SDN could flexibly program the network for statistical QoS requirements of different mobile users.

H. Security and Privacy

Security and privacy are significant concerns in IoT due to the Heterogeneity of IoT applications, different interfaces, and implementations. Fig. 9 shows SDIC-IoT vulnerabilities against various attacks. Flooding attacks on communication channels between A-CPI and C-DPI can bring the SDIC-IoT network down. Attack possibilities include distributed denial of service attacks, link flooding attacks, man in the middle attacks, and privacy breaching attacks [71]. Attackers can attack communication channels between different planes. For the successful implementation of an IoT application, a

secure medium is of the utmost priority. Most IoT devices are always connected to the Internet. The resource-limited nature of IoT makes it vulnerable to being used as attack sources. The ability to generate network traffic makes IoT a readily available source for flooding attacks. The application plane of SDIC-IoT could be used to develop novel applications against security and privacy-breaching attacks. The capability of the controller to extract network statistics could be used to develop pattern analysis-based security solutions [135], [136]. Moreover, IoT devices use different operating systems, making it challenging to patch security updates in all the devices.

VII. CONCLUSION

IoT uses communication technologies to connect physical objects with the Internet to realize a wide variety of services. IoT services challenge current networks due to heterogeneity, a higher number of devices, and diverse implementations. SDN and ICN are two complementary technologies, which tackle different networking challenges to deliver IoT services. SDN ensures the programmability of the network by implementing virtualized network functions over general-purpose hardware. It gives the flexibility of implementing ICN functionality of in-network caching and name-based routing in an SDN-controlled environment. Thus, the SDICN paradigm provides an effective solution to support IoT services.

In this paper, we covered guidelines to integrate SDIC-IoT for IoT service provisioning. This paper analyzed the motivation of SDIC-IoT integration and demonstrated its benefits for IoT service provisioning. We discussed critical requirements in envisioning the SDIC-IoT architecture while considering resource-limitation constraints, content-centric nature, and massive repository of IoT infrastructure. We described the taxonomies of existing approaches from several aspects such as caching, mobility, traffic engineering, implementation, and virtualization. Furthermore, we discussed several use case implementations of SDIC-IoT, such as smart cities, smart homes, smart grids, smart healthcare, and smart transportation that provide guidelines for future SDIC-IoT systems. Finally, we presented current challenges that impact SDIC-IoT implementations and future research directions.

REFERENCES

- [1] W. Rafique, L. Qi, I. Yaqoob, M. Imran, R. U. Rasool, and W. Dou, "Complementing IoT services through software defined networking and edge computing: A comprehensive survey," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 3, pp. 1761–1804, 2020.
- [2] B. Nour, S. Mastorakis, R. Ullah, and N. Stergiou, "Information-centric networking in wireless environments: Security risks and challenges," *IEEE Wireless Communications*, vol. 28, no. 2, pp. 121–127, 2021.
- [3] M. Amadeo, C. Campolo, and A. Molinaro, "Information-centric networking for connected vehicles: a survey and future perspectives," *IEEE Communications Magazine*, vol. 54, no. 2, pp. 98–104, 2016.
- [4] S. Al-Sarawi, M. Anbar, K. Alieyan, and M. Alzubaidi, "Internet of things (IoT) communication protocols," in *Proc. 8th Int. Conf. on information technology (ICIT)*, 2017, pp. 685–690.
- [5] W. Bekri, R. Jmal, and L. Chaari Fourati, "Internet of things management based on software defined networking: a survey," *International Journal of Wireless Information Networks*, vol. 27, pp. 385–410, 2020.
- [6] L. Bracciale, P. Loreti, A. Detti, R. Paolillo, and N. B. Melazzi, "Lightweight named object: An ICN-based abstraction for IoT device programming and management," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 5029–5039, 2019.
- [7] B. Nour, K. Sharif, F. Li, H. Mounsla, and Y. Liu, "A unified hybrid information-centric naming scheme for IoT applications," *Computer Communications*, vol. 150, pp. 103–114, 2020.
- [8] S. Arshad, M. A. Azam, S. H. Ahmed, and J. Loo, "Towards information-centric networking (ICN) naming for Internet of things (IoT) the case of smart campus," in *Proc. Int. Conf. on Future Networks and Distributed Systems*, 2017, pp. 1–6.
- [9] S. Mastorakis, A. Mtibaa, J. Lee, and S. Misra, "Icedge: When edge computing meets information-centric networking," *IEEE Internet of Things Journal*, vol. 7, no. 5, pp. 4203–4217, 2020.
- [10] Y. Thomas, N. Fotiou, S. Toumpis, and G. C. Polyzos, "Improving mobile ad hoc networks using hybrid ip-information centric networking," *Computer Communications*, vol. 156, pp. 25–34, 2020.
- [11] B. K. Mohanta, D. Jena, U. Satapathy, and S. Patnaik, "Survey on IoT security: challenges and solution using machine learning, artificial intelligence and blockchain technology," *Internet of Things*, p. 100227, 2020.
- [12] B. Wissingh, C. A. Wood, A. Afanasyev, L. Zhang, D. Oran, and C. F. Tschudin, "Information-centric networking (ICN): Content-centric networking (ccnx) and named data networking (ndn) terminology," *RFC*, vol. 8793, pp. 1–17, 2020.
- [13] S. Tayeb, S. Latifi, and Y. Kim, "A survey on IoT communication and computation frameworks: An industrial perspective," in *Proc. IEEE 7th annual Computing and Communication Workshop and Conf. (CCWC)*, 2017, pp. 1–6.
- [14] Q.-Y. Zhang, X.-W. Wang, M. Huang, K.-Q. Li, and S. K. Das, "Software defined networking meets information centric networking: A survey," *IEEE Access*, vol. 6, pp. 39 547–39 563, 2018.
- [15] C. Gündoğan, P. Kietzmann, T. C. Schmidt, and M. Wählisch, "Information-centric networking for the industrial Internet of things," in *Proc. Wireless Networks and Industrial IoT*. Springer, 2021, pp. 171–189.
- [16] S. Arshad, M. A. Azam, M. H. Rehmani, and J. Loo, "Recent advances in information-centric networking-based Internet of things ICN-IoT," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 2128–2158, 2018.
- [17] F. Al-Turjman, E. Ever, Y. B. Zikria, S. W. Kim, and A. Elmahgoubi, "Sahci: scheduling approach for heterogeneous content-centric IoT applications," *IEEE Access*, vol. 7, pp. 80 342–80 349, 2019.
- [18] I. U. Din, S. Hassan, A. Almogren, F. Ayub, and M. Guizani, "PUC: Packet update caching for energy efficient IoT-based information-centric networking," *Future Generation Computer Systems*, vol. 111, pp. 634–643, 2020.
- [19] Y. Hua, L. Guan, and K. G. Kyriakopoulos, "A fog caching scheme enabled by ICN for IoT environments," *Future Generation Computer Systems*, vol. 111, pp. 82–95, 2020.
- [20] Q. Zhang, X. Wang, J. Lv, and M. Huang, "Intelligent content-aware traffic engineering for SDN: An ai-driven approach," *IEEE Network*, vol. 34, no. 3, pp. 186–193, 2020.
- [21] O. Salman, I. Elhajj, A. Chehab, and A. Kayssi, "IoT survey: An SDN and fog computing perspective," *Computer Networks*, vol. 143, pp. 221–246, 2018.
- [22] C. Fang, H. Yao, Z. Wang, W. Wu, X. Jin, and F. R. Yu, "A survey of mobile information-centric networking: Research issues and challenges," *IEEE Communications Surveys & Tutorials*, vol. 20, no. 3, pp. 2353–2371, 2018.
- [23] C. Fang, F. R. Yu, T. Huang, J. Liu, and Y. Liu, "A survey of energy-efficient caching in information-centric networking," *IEEE Communications Magazine*, vol. 52, no. 11, pp. 122–129, 2014.
- [24] M. Zhang, H. Luo, and H. Zhang, "A survey of caching mechanisms in information-centric networking," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 3, pp. 1473–1499, 2015.
- [25] I. Farris, T. Taleb, Y. Khettab, and J. Song, "A survey on emerging SDN and NFV security mechanisms for IoT systems," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 1, pp. 812–837, 2018.
- [26] N. Bizanis and F. A. Kuipers, "SDN and virtualization solutions for the Internet of things: A survey," *IEEE Access*, vol. 4, pp. 5591–5606, 2016.
- [27] R. Jmal and L. C. Fourati, "Content-centric networking management based on software defined networks: Survey," *IEEE Transactions on network and service management*, vol. 14, no. 4, pp. 1128–1142, 2017.
- [28] F. Bannour, S. Souihi, and A. Mellouk, "Adaptive distributed SDN controllers: Application to content-centric delivery networks," *Future Generation Computer Systems*, vol. 113, pp. 78–93, 2020.
- [29] W. Shang, Y. Yu, R. Droms, and L. Zhang, "Challenges in IoT networking via tcp/ip architecture," *NDN Project*, 2016.

- [30] T. Guesmi, A. Kalghoum, B. M. Alshammari, H. Alsaif, and A. Alzamil, "Leveraging software-defined networking approach for future information-centric networking enhancement," *Symmetry*, vol. 13, no. 3, p. 441, 2021.
- [31] A. Raghuvanshi, U. K. Singh, M. Shuaib, and S. Alam, "An investigation of various applications and related security challenges of Internet of things," *Materials Today: Proceedings*, 2021.
- [32] H. Saadeh, W. Almobaideen, K. E. Sabri, and M. Saadeh, "Hybrid SDN-ICN architecture design for the Internet of things," in *Proc. 6th Int. Conf. on Software Defined Systems (SDS)*, 2019, pp. 96–101.
- [33] P. P. Ray and N. Kumar, "SDN/NFV architectures for edge-cloud oriented IoT: A systematic review," *Computer Communications*, 2021.
- [34] R. Jmal and L. C. Fourati, "Emerging applications for future Internet approach based-on SDN and ICN," in *Proc. IEEE/ACS 14th Int. Conf. on Computer Systems and Applications (AICCSA)*, 2017, pp. 208–213.
- [35] K. Kaur and R. Kapoor, "Mcpn: multi-hop clustering protocol using cache nodes in wsn," *Wireless Personal Communications*, vol. 109, no. 3, pp. 1727–1745, 2019.
- [36] O. Serhane, K. Yahyaoui, B. Nour, and H. Moun gla, "A survey of ICN content naming and in-network caching in 5g and beyond networks," *IEEE Internet of Things Journal*, 2020.
- [37] M. Amadeo, C. Campolo, G. Ruggeri, A. Molinaro, and A. Iera, "Understanding name-based forwarding rules in software-defined named data networking," in *Proc. IEEE Int. Conf. on Communications (ICC)*, 2020, pp. 1–6.
- [38] D. R. Cheriton and M. Gritter, "A scalable deployable nat-based Internet architecture," *Stanford Computer Science Technical Report*, 2000.
- [39] "National science foundation of the usa," Date Accessed: June 06, 2021. [Online]. Available: National science foundation (NSF) future of Internet architecture (FIA) program. <http://www.nets-fia.net/>.
- [40] T. Koponen, M. Chawla, B.-G. Chun, A. Ermolinskiy, K. H. Kim, S. Shenker, and I. Stoica, "A data-oriented (and beyond) network architecture," in *Proc. 2007 Conf. on Applications, technologies, architectures, and protocols for computer communications*, 2007, pp. 181–192.
- [41] A. Köpsel and H. Woesner, "Ofelia—pan-european test facility for openflow experimentation," in *Proc. Springer European Conf. on a Service-Based Internet*, 2011, pp. 311–312.
- [42] B. Ahlgren, C. Dannewitz, C. Imbrenda, D. Kutscher, and B. Ohlman, "A survey of information-centric networking," *IEEE Communications Magazine*, vol. 50, no. 7, pp. 26–36, 2012.
- [43] G. Carofiglio, M. Gallo, and L. Muscariello, "Icp: Design and evaluation of an interest control protocol for content-centric networking," in *Proc. IEEE INFOCOM Workshops*, 2012, pp. 304–309.
- [44] A. Detti, N. Blefari Melazzi, S. Salsano, and M. Pomposini, "Conet: a content centric inter-networking architecture," in *Proc. ACM SIGCOMM workshop on Information-centric networking*, 2011, pp. 50–55.
- [45] M. Amadeo, A. Molinaro, and G. Ruggeri, "E-chanet: Routing, forwarding and transport in information-centric multihop wireless networks," *Computer communications*, vol. 36, no. 7, pp. 792–803, 2013.
- [46] G. Xylomenos, C. N. Ververidis, V. A. Siris, N. Fotiou, C. Tsilopoulos, X. Vasilakos, K. V. Katsaros, and G. C. Polyzos, "A survey of information-centric networking research," *IEEE communications surveys & tutorials*, vol. 16, no. 2, pp. 1024–1049, 2013.
- [47] D. Trossen, M. J. Reed, J. Riihijärvi, M. Georgiades, N. Fotiou, and G. Xylomenos, "Ip over ICN-the better ip?" in *Proc. European Conf. on networks and communications (EuCNC)*, 2015, pp. 413–417.
- [48] A. Rahman, D. Trossen, D. Kutscher, and R. Ravindran, "Deployment considerations for information-centric networking (ICN)," *ICNRG draft*, 2018.
- [49] M. Vahlenkamp, F. Schneider, D. Kutscher, and J. Seedorf, "Enabling ICN in ip networks using SDN," in *Proc. 21st IEEE Int. Conf. on Network Protocols (ICNP)*, 2013, pp. 1–2.
- [50] Z. Sheng, S. Yang, Y. Yu, A. V. Vasilakos, J. A. McCann, and K. K. Leung, "A survey on the ietf protocol suite for the Internet of things: Standards, challenges, and opportunities," *IEEE wireless communications*, vol. 20, no. 6, pp. 91–98, 2013.
- [51] S. Debroy, P. Samanta, A. Bashir, and M. Chatterjee, "SpEED-IoT: Spectrum aware energy efficient routing for device-to-device IoT communication," *Future Generation Computer Systems*, vol. 93, pp. 833–848, 2019.
- [52] L. Atzori, A. Iera, G. Morabito, and M. Nitti, "The social Internet of things (SIoT)—when social networks meet the Internet of things: Concept, architecture and network characterization," *Computer networks*, vol. 56, no. 16, pp. 3594–3608, 2012.
- [53] G. J. Joyia, R. M. Liaqat, A. Farooq, and S. Rehman, "Internet of medical things (iomt): applications, benefits and future challenges in healthcare domain," *J Commun*, vol. 12, no. 4, pp. 240–247, 2017.
- [54] J. A. Khan, C. Westphal, and Y. Ghamri-Doudane, "Information-centric fog network for incentivized collaborative caching in the Internet of everything," *IEEE Communications Magazine*, vol. 57, no. 7, pp. 27–33, 2019.
- [55] C. Gündoğan, C. Amsüss, T. C. Schmidt, and M. Wählisch, "Toward a restful information-centric web of things: a deeper look at data orientation in coap," in *Proc. 7th ACM Conf. on Information-Centric Networking*, 2020, pp. 77–88.
- [56] S. L. Levin and S. Schmidt, "Ipv4 to ipv6: Challenges, solutions, and lessons," *Telecommunications Policy*, vol. 38, no. 11, pp. 1059–1068, 2014.
- [57] B. Nour, K. Sharif, F. Li, and H. Moun gla, "A distributed ICN-based IoT network architecture: An ambient assisted living application case study," in *Proc. IEEE Global Communications Conf.*, 2017, pp. 1–6.
- [58] A. Ravi, P. Ramanathan, and K. M. Sivalingam, "Integrated network coding and caching in information-centric networks: revisiting pervasive caching in the ICN framework," *Photonic Network Communications*, vol. 30, no. 3, pp. 416–427, 2015.
- [59] M. Amadeo, C. Campolo, A. Iera, and A. Molinaro, "Information centric networking in IoT scenarios: The case of a smart home," in *Proc. IEEE Int. Conf. on communications (ICC)*, 2015, pp. 648–653.
- [60] S. Salsano, N. Blefari-Melazzi, A. Detti, G. Morabito, and L. Veltri, "Information centric networking over SDN and openflow: Architectural aspects and experiments on the ofelia testbed," *Computer Networks*, vol. 57, no. 16, pp. 3207–3221, 2013.
- [61] R. U. Rasool, K. Ahmed, Z. Anwar, H. Wang, U. Ashraf, and W. Rafique, "Cyberpulse++: A machine learning-based security framework for detecting link flooding attacks in software defined networks," *International Journal of Intelligent Systems*, 2021.
- [62] W. Rafique, X. He, Z. Liu, Y. Sun, and W. Dou, "Cfdefense: A security solution to detect and mitigate crossfire attacks in software-defined IoT-edge infrastructure," in *Proc. IEEE 21st Int. Conf. on High Performance Computing and Communications; IEEE 17th Int. Conf. on Smart City; IEEE 5th Int. Conf. on Data Science and Systems (HPCC/SmartCity/DSS)*, 2019, pp. 500–509.
- [63] Z. Eghbali and M. Z. Lighvan, "A hierarchical approach for accelerating IoT data management process based on SDN principles," *Journal of Network and Computer Applications*, vol. 181, p. 103027, 2021.
- [64] S. Bera, S. Misra, S. K. Roy, and M. S. Obaidat, "Soft-wsn: Software-defined wsn management system for IoT applications," *IEEE Systems Journal*, vol. 12, no. 3, pp. 2074–2081, 2016.
- [65] Z.-Y. Ai, F. Song, and X. Wang, "Combining SDN and ICN for network survivability improvement," *J. Internet Serv. Inf. Secur.*, vol. 8, no. 1, pp. 18–30, 2018.
- [66] M. Arumaiturai, J. Chen, E. Monticelli, X. Fu, and K. K. Ramakrishnan, "Exploiting ICN for flexible management of software-defined networks," in *Proc. 1st ACM Conf. on Information-Centric Networking*, 2014, pp. 107–116.
- [67] C. Kolias, G. Kambourakis, A. Stavrou, and J. Voas, "DDoS in the IoT: Mirai and other botnets," *Computer*, vol. 50, no. 7, pp. 80–84, 2017.
- [68] P. Nikander, J. Arkko, T. Aura, G. Montenegro, and E. Nordmark, "Mobile ip version 6 route optimization security design background," *draft-ietf-mip6-ro-sec-03 (work in progress)*, 2005.
- [69] S. Bhunia, S. Ray, and S. Sur-Kolay, *Fundamentals of IP and SoC security*. Springer, 2017.
- [70] S. Pirbhulal, W. Wu, K. Muhammad, I. Mehmood, G. Li, and V. H. C. de Albuquerque, "Mobility enabled security for optimizing IoT based intelligent applications," *IEEE Network*, vol. 34, no. 2, pp. 72–77, 2020.
- [71] W. Rafique, M. Khan, N. Sarwar, and W. Dou, "A security framework to protect edge supported software defined Internet of things infrastructure," in *Proc. Springer Int. Conf. on Collaborative Computing: Networking, Applications and Worksharing*, 2019, pp. 71–88.
- [72] J. Li, K. Xue, J. Liu, Y. Zhang, and Y. Fang, "An ICN/SDN-based network architecture and efficient content retrieval for future satellite-terrestrial integrated networks," *IEEE Network*, vol. 34, no. 1, pp. 188–195, 2019.
- [73] P. Benedetti, G. Piro, and L. A. Grieco, "An energy efficient and software-defined information-centric networking approach to consumer mobility," in *Proc. 22nd Int. Conf. on Transparent Optical Networks (ICTON)*, 2020, pp. 1–4.
- [74] J. McCarthy, S. Chaudhry, P. Kuppuudaiyar, R. Loomba, and S. Clarke, "QoSA-ICN: An information-centric approach to qos in vehicular environments," *Vehicular Communications*, vol. 30, p. 100351, 2021.

- [75] M. Hussaini, M. A. Naeem, and B.-S. Kim, "Opmss: Optimal producer mobility support solution for named data networking," *Applied Sciences*, vol. 11, no. 9, p. 4064, 2021.
- [76] W. Yang, Y. Qin, and B. Wu, "A hyperbolic routing scheme for information-centric Internet of things with edge computing," *Wireless Networks*, pp. 1–13, 2021.
- [77] T. Das and M. Gurusamy, "Multi-objective control plane dimensioning in hybrid SDN/legacy networks," *IEEE Transactions on Network and Service Management*, 2021.
- [78] H.-C. Hsieh, M.-L. Chiang, and T.-Y. Chang, "Improving the fault-tolerance of software-defined networks with dynamic overlay agreement," *Cluster Computing*, pp. 1–18, 2021.
- [79] L. Veltri, G. Morabito, S. Salsano, N. Blefari-Melazzi, and A. Detti, "Supporting information-centric functionality in software defined networks," in *Proc. IEEE Int. Conf. on Communications (ICC)*, 2012, pp. 6645–6650.
- [80] "Sd-icn."
- [81] S. Eum, M. Jibiki, M. Murata, H. Asaeda, and N. Nishinaga, "A design of an ICN architecture within the framework of SDN," in *Proc. 7th Int. Conf. on Ubiquitous and Future Networks*, 2015, pp. 141–146.
- [82] M. Arumathurai, J. Chen, E. Maiti, X. Fu, and K. Ramakrishnan, "Prototype of an ICN based approach for flexible service chaining in SDN," in *Proc. IEEE Conf. on Computer Communications Workshops (INFOCOM WKSHPS)*, 2015, pp. 5–6.
- [83] W. Xiulei, C. Ming, H. Chao, W. Xi, and X. Changyou, "SDICN: A software defined deployable framework of information centric networking," *China Communications*, vol. 13, no. 3, pp. 53–65, 2016.
- [84] G. Siracusano, S. Salsano, P. L. Ventre, A. Detti, O. Rashed, and N. Blefari-Melazzi, "A framework for experimenting ICN over SDN solutions using physical and virtual testbeds," *Computer Networks*, vol. 134, pp. 245–259, 2018.
- [85] Z. Liu, J. Zhu, C. Pan, and G. Song, "Satellite network architecture design based on SDN and ICN technology," in *Proc. 8th Int. Conf. on Electronics Information and Emergency Communication (ICEIEC)*, 2018, pp. 124–131.
- [86] F. Ren, Y. Qin, H. Zhou, and Y. Xu, "Mobility management scheme based on software defined controller for content-centric networking," in *Proc. IEEE Conf. on Computer Communications Workshops (INFOCOM WKSHPS)*, 2016, pp. 193–198.
- [87] Y. Zhang and Y. Wang, "SDN based ICN architecture for the future integration network," in *Proc. 16th International Symposium on Communications and Information Technologies (ISCIT)*, 2016, pp. 474–478.
- [88] C. Xing, K. Ding, C. Hu, M. Chen, and B. Xu, "SD-ICN: Toward wide area deployable software defined information centric networking," *KSI Transactions on Internet and Information Systems (TIIS)*, vol. 10, no. 5, pp. 2267–2285, 2016.
- [89] R. Ravindran, A. Chakraborti, S. O. Amin, A. Azgin, and G. Wang, "5g-ICN: Delivering ICN services over 5g using network slicing," *IEEE Communications Magazine*, vol. 55, no. 5, pp. 101–107, 2017.
- [90] Y. He, C. Liang, Z. Zhang, F. R. Yu, N. Zhao, H. Yin, and Y. Zhang, "Resource allocation in software-defined and information-centric vehicular networks with mobile edge computing," in *Proc. IEEE 86th Vehicular Technology Conf. (VTC-Fall)*, 2017, pp. 1–5.
- [91] G. M. Leal, I. Zacarias, J. M. Stocchero, and E. P. de Freitas, "Empowering command and control through a combination of information-centric networking and software defined networking," *IEEE Communications Magazine*, vol. 57, no. 8, pp. 48–55, 2019.
- [92] Y. Ban, "Software-defined information-centric networking based exercise intensity evaluation of volleyball player: An efficient convolutional neural network method," *Internet Technology Letters*, 2020.
- [93] D. Raychaudhuri, K. Nagaraja, and A. Venkataramani, "Mobilityfirst: a robust and trustworthy mobility-centric architecture for the future Internet," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 16, no. 3, pp. 2–13, 2012.
- [94] P. Li, W. Muqing, W. Ning, and L. Hongbao, "Supporting information-centric networking in SDN," *International Journal of Future Computer and Communication*, vol. 4, no. 6, p. 386, 2015.
- [95] A. El Mougy, "On the integration of software-defined and information-centric networking paradigms," in *Proc. IEEE Int. Symposium on Signal Processing and Information Technology (ISSPIT)*, 2015, pp. 105–110.
- [96] S. Gao, Y. Zeng, H. Luo, and H. Zhang, "Scalable control plane for intra-domain communication in software defined information centric networking," *Future Generation Computer Systems*, vol. 56, pp. 110–120, 2016.
- [97] G. Petropoulos, K. V. Katsaros, and M.-E. Xezonaki, "Openflow-compliant topology management for SDN-enabled information centric networks," in *Proc. IEEE Symposium on Computers and Communications (ISCC)*, 2017, pp. 951–954.
- [98] V. G. Vassilakis, I. D. Moscholios, B. A. Alzahrani, and M. D. Logothetis, "Toward information-centric software-defined cellular networks," in *Proc. 14th Int. Conf. on Telecommunications (ConTEL)*, 2017, pp. 99–106.
- [99] A. Kalghoum and S. M. Gammar, "Towards new information centric networking strategy based on software defined networking," in *Proc. IEEE Wireless Communications and Networking Conf. (WCNC)*, 2017, pp. 1–6.
- [100] M. Amadeo, C. Campolo, G. Ruggeri, A. Molinaro, and A. Iera, "Towards software-defined fog computing via named data networking," in *Proc. IEEE Conf. on Computer Communications Workshops (INFOCOM WKSHPS)*, 2019, pp. 133–138.
- [101] X. Zhang and Q. Zhu, "Information-centric network virtualization for qos provisioning over software defined wireless networks," in *Proc. MILCOM IEEE Military Communications Conf.*, 2016, pp. 1028–1033.
- [102] J. Li, J. Wu, G. Xu, J. Li, X. Zheng, and A. Jolfaei, "Integrating NFV and ICN for advanced driver-assistance systems," *IEEE Internet of Things Journal*, vol. 7, no. 7, pp. 5861–5873, 2019.
- [103] S. Liao, J. Wu, J. Li, A. K. Bashir, S. Mumtaz, A. Jolfaei, and N. Kvedaraite, "Cognitive popularity based ai service sharing for software-defined information-centric networks," *IEEE Transactions on Network Science and Engineering*, vol. 7, no. 4, pp. 2126–2136, 2020.
- [104] D. Syrivelis, G. Parisi, D. Trossen, P. Flegkas, V. Sourlas, T. Korakis, and L. Tassioulas, "Pursuing a software defined information-centric network," in *Proc. European Workshop on Software Defined Networking*, 2012, pp. 103–108.
- [105] H. Huang, H. Yin, G. Min, H. Jiang, J. Zhang, and Y. Wu, "Data-driven information plane in software-defined networking," *IEEE Communications Magazine*, vol. 55, no. 6, pp. 218–224, 2017.
- [106] H. Luo, J. Cui, Z. Chen, M. Jin, and H. Zhang, "Efficient integration of software defined networking and information-centric networking with color," in *Proc. IEEE Global Communications Conf.*, 2014, pp. 1962–1967.
- [107] X. Zhang and Q. Zhu, "Information-centric virtualization for software-defined statistical qos provisioning over 5g multimedia big data wireless networks," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 8, pp. 1721–1738, 2019.
- [108] A. Tariq, R. A. Rehman, and B.-S. Kim, "Epf—an efficient forwarding mechanism in SDN controller enabled named data IoTs," *Applied Sciences*, vol. 10, no. 21, p. 7675, 2020.
- [109] W.-X. Liu, J. Zhang, Z.-W. Liang, L.-X. Peng, and J. Cai, "Content popularity prediction and caching for ICN: A deep learning approach with SDN," *IEEE access*, vol. 6, pp. 5075–5089, 2017.
- [110] J. Liu, Z. Hu, J. Guo, and H. Hao, "An Internet application-driven cache placement algorithm for software-defined information-centric networking," in *Proc. Int. Conf. on Computer Technology, Electronics and Communication (ICCTEC)*, 2017, pp. 1374–1378.
- [111] F. Al-Turjman, "Fog-based caching in software-defined information-centric networks," *Computers & Electrical Engineering*, vol. 69, pp. 54–67, 2018.
- [112] E. B. Nascimento, D. D. de Macedo, E. D. Moreno, L. C. E. De Bona, and M. A. Capretz, "Evaluation of cache for bandwidth optimization in ICN through software-defined networks," in *Proc IEEE Symposium on Computers and Communications (ISCC)*, 2018, pp. 00656–00661.
- [113] F. Xu, F. Yang, S. Bao, and C. Zhao, "Dqn inspired joint computing and caching resource allocation approach for software defined information-centric Int.net of things network," *IEEE Access*, vol. 7, pp. 61987–61996, 2019.
- [114] Z. Zhang, C.-H. Lung, M. St-Hilaire, and I. Lambadaris, "An SDN-based caching decision policy for video caching in information-centric networking," *IEEE Transactions on Multimedia*, vol. 22, no. 4, pp. 1069–1083, 2019.
- [115] A. Chanda, C. Westphal, and D. Raychaudhuri, "Content based traffic engineering in software defined information centric networks," in *Proc. IEEE Conf. on Computer Communications Workshops (INFOCOM WKSHPS)*, 2013, pp. 357–362.
- [116] J. Chen, S. Li, H. Yu, Y. Zhang, D. Raychaudhuri, R. Ravindran, H. Gao, L. Dong, G. Wang, and H. Liu, "Exploiting ICN for realizing service-oriented communication in IoT," *IEEE Communications Magazine*, vol. 54, no. 12, pp. 24–30, 2016.
- [117] J. Lv, X. Wang, M. Huang, J. Shi, K. Li, and J. Li, "RISC: ICN routing mechanism incorporating SDN and community division," *Computer Networks*, vol. 123, pp. 88–103, 2017.

- [118] Y. Cai, J. Liu, X. Fan, Y. Qiu, and B. Tan, "Software defined status aware routing in content-centric networking," in *Proc. Int. Conf. on Information Networking (ICOIN)*, 2018, pp. 283–288.
- [119] Y. He, F. R. Yu, N. Zhao, V. C. Leung, and H. Yin, "Software-defined networks with mobile edge computing and caching for smart cities: A big data deep reinforcement learning approach," *IEEE Communications Magazine*, vol. 55, no. 12, pp. 31–37, 2017.
- [120] A. Fekih, S. Gaied, and H. Yousef, "A comparative study of content-centric and software defined networks in smart cities," in *Proc. Int. Conf. on Smart, Monitored and Controlled Cities (SM2C)*, 2017, pp. 147–151.
- [121] M. A. Naeem, R. Ali, B.-S. Kim, S. A. Nor, and S. Hassan, "A periodic caching strategy solution for the smart city in information-centric Internet of things," *Sustainability*, vol. 10, no. 7, p. 2576, 2018.
- [122] J. Burke, P. Gasti, N. Nathan, and G. Tsudik, "Securing instrumented environments over content-centric networking: The case of lighting control and ndn," in *Proc. IEEE Conf. on Computer Communications Workshops (INFOCOM WKSHPS)*, 2013, pp. 394–398.
- [123] M. A. Hail and S. Fischer, "IoT for aal: An architecture via information-centric networking," in *Proc. IEEE Globecom Workshops (GC Wkshps)*, 2015, pp. 1–6.
- [124] J. Zhang, Q. Li, and E. M. Schooler, "ihems: An information-centric approach to secure home energy management," in *Proc. IEEE 3rd Int. Conf. on Smart Grid Communications (SmartGridComm)*, 2012, pp. 217–222.
- [125] W. Han, M. Dong, K. Ota, J. Wu, J. Li, and G. Li, "Sd-opts: Software-defined on-path time synchronization for information-centric smart grid," in *Proc. IEEE Global Communications Conf.*, 2017, pp. 1–6.
- [126] K. V. Katsaros, W. K. Chai, N. Wang, G. Pavlou, H. Bontius, and M. Paolone, "Information-centric networking for machine-to-machine data delivery: a case study in smart grid applications," *IEEE Network*, vol. 28, no. 3, pp. 58–64, 2014.
- [127] S. H. Bouk, S. H. Ahmed, D. Kim, and H. Song, "Named-data-networking-based its for smart cities," *IEEE Communications Magazine*, vol. 55, no. 1, pp. 105–111, 2017.
- [128] D. Saxena, V. Raychoudhury, and C. Becker, "Implementation and performance evaluation of name-based forwarding schemes in v-ndn," in *Proc. 18th Int. Conf. on Distributed Computing and Networking*, 2017, pp. 1–4.
- [129] E. Fouladi, H. R. Baghaee, M. Bagheri, and G. Gharehpetian, "Power management of microgrids including phevs based on maximum employment of renewable energy resources," *IEEE Transactions on Industry Applications*, vol. 56, no. 5, pp. 5299–5307, 2020.
- [130] G. S. Aujla, A. Jindal, and N. Kumar, "Evaas: Electric vehicle-as-a-service for energy trading in SDN-enabled smart transportation system," *Computer Networks*, vol. 143, pp. 247–262, 2018.
- [131] D. Saxena and V. Raychoudhury, "Design and verification of an ndn-based safety-critical application: A case study with smart healthcare," *IEEE transactions on systems, man, and cybernetics: systems*, vol. 49, no. 5, pp. 991–1005, 2017.
- [132] K. Cho, M. Lee, K. Park, T. T. Kwon, Y. Choi, and S. Pack, "Wave: Popularity-based and collaborative in-network caching for content-oriented networks," in *Proc. IEEE INFOCOM Workshops*, 2012, pp. 316–321.
- [133] H. Nakayama, S. Ata, and I. Oka, "Caching algorithm for content-oriented networks using prediction of popularity of contents," in *Proc. IFIP/IEEE International Symposium on Integrated Network Management (IM)*, 2015, pp. 1171–1176.
- [134] Z. Xiaoqiang, Z. Min, and W. Muqing, "An in-network caching scheme based on betweenness and content popularity prediction in content-centric networking," in *Proc. IEEE 27th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, 2016, pp. 1–6.
- [135] Z. Abou El Houda, A. Hafid, and L. Khoukhi, "Co-IoT: A collaborative DDoS mitigation scheme in IoT environment based on blockchain using SDN," in *Proc. IEEE Global Communications Conf. (GLOBECOM)*, 2019, pp. 1–6.
- [136] Z. Abou El Houda, A. S. Hafid, and L. Khoukhi, "Cochain-SC: An intra-and inter-domain DDoS mitigation scheme based on blockchain using SDN and smart contract," *IEEE Access*, vol. 7, pp. 98 893–98 907, 2019.