Robust Proactive Mobility Management in Named Data Networking Under Erroneous Content Prediction

Hesham Farahat
ECE Department, Queen's University
Kingston, ON, Canada
h.farahat@queensu.ca

Ramy Atawia
ECE Department, Queen's University
Kingston, ON, Canada
ramy.atawia@queensu.ca

Hossam S. Hassanein
School of Computing, Queen's University
Kingston, ON, Canada
hossam@cs.queensu.ca

***Abstract*—Named Data Networking (NDN) is a promising paradigm for the future Internet to survive the growing data demand. Supporting seamless operation during user mobility is one of the main challenges in NDN. In this paper, we investigate optimal caching for *producer* mobility under prediction uncertainties. Mainly, we propose a *stochastic* optimization framework that exploits location and data requests' predictors to cache data proactively before handover. We model the problem using Chance Constraint Programming (CCP) that *probabilistically* incorporates the uncertainty in data prediction and models the trade-off between network overhead and *Consumer* satisfaction. A deterministic formulation is derived to obtain a closed form Integer Linear Programming model based on the prediction error model. The proposed framework is then implemented in ndnSIM and Gurobi, and simulation experiments are conducted to provide benchmark solutions for *robust proactive* caching. The results show that such *robust* scheme satisfies the *consumers'* quality of experience under imperfect prediction of future content requested from mobile *producers*. Hence, sustains the prediction gains over conventional non-predictive schemes without compromising the network overhead. We believe that such results drive incentives for deploying *proactive* mobility management in future NDN.**

I. Introduction

The ongoing evolution of technology in devices and data content has considerably changed the way we use the Internet. With the proliferation of services such as Facebook and Youtube, among others, users can share data hungry content (e.g. live videos) while moving. Cisco has predicted in 2017's Virtual Network Index that the amount of traffic will increase threefold by 2020, forming a 100-fold increase from 2005 to 2020 [1]. However, the Internet was primarily designed to be a network of *hosts* which is not able to scale up with future data usage [2].

This led to a new promising paradigm for future Internet referred to as Information-Centric Network (ICN) [3]. In essence, ICN considers *content*, instead of *hosts*, as the building block of the network, which provides many features such as data-to-location decoupling and automatic content caching . Among the paradigms of ICN is Named Data Network (NDN) [2] which is one of the four projects under NSF's Future Internet Architecture Program.

This research is supported by the Natural Sciences and Engineering Research Council of Canada (NSERC) under grant number: STPGP 479248.

One of the main challenges of NDN is supporting seamless user mobility. Current NDN designs depend on *reactive* mobility management schemes that retransmit *interests* in case of content unavailability due to mobility [4]. However, such simple management does not guarantee seamless content delivery, and result in suboptimal utilization of network resources. A new mobility design, for both data *producers* or *consumers*, is thus essential to satisfy stringent delay and throughput requirements by the NDN *consumers*. The mobility of *consumers* was previously analyzed in [5] and showed that there is a performance gap between reactive and proactive schemes. In this work, we focus on *producer* mobility that must be handled explicitly as the content providers are expected to be mobile users. One of the promising approaches to support seamless *producer* mobility is proactive caching where network resources are used to store predicted future *interest* requests. This approach was evaluated in our prior work [6] under *idealistic* conditions where perfect prediction was assumed. Compared to conventional non-predictive schemes, such proactive support of *producer* mobility results in satisfying delay requirements of *consumers* with limited overhead.

In this paper, we support the practicality of proactive mobility management by considering uncertainty in predicted requests. Frequent changes in user demands or erroneous previous information will typically result in imperfect prediction of future *interest* requests. Thus, the cached content will not be used, resulting in suboptimal utilization of caching resources. This is in addition to compromising *consumers'* quality of experience due to missing the true content requested during *producer* handover.

We introduce a *stochastic* optimization framework for proactive *producer* mobility management that is *robust* to content prediction errors. In addition to caching the most probable future content, the framework adopts prediction errors model to cache other less probable content that might be requested by the users. Thus, avoids an increased *consumer* delay if such less probable content is not cached in advance. We summarize

978-1-5090-5019-2/17/$31.00 ©2017 IEEE

our contributions as follows:

1) We formulate the *producer* mobility management problem using Chance Constraint Programming (CCP). In particular, predicted *interest* requests are modeled as random variables in order to incorporate the associated uncertainty. The CCP will bound the satisfaction degree of network constraints by a certain probabilistic level. Thus, it allows network designers to strike a balance between caching costs and overhead, on the one hand, and the risk of violating *consumer*'s delay, on the other hand, during prediction uncertainties.
2) We derive a deterministic equivalent form for the *probabilistic* CCP model using Scenario Approximation (SA). In essence, the Probability Mass Function (PMF) of the predicted *interests* is used to replace the random variables, resulting in a closed form Integer Linear Programming (ILP) model that can be solved by numerical optimization. This enables deriving benchmark solutions for future *robust* predictive schemes and redefines the prediction gains under real-world uncertainties.

The remainder of the paper is organized as follows. In Section II, a background of NDN and its mobility management is given, and followed by a background on stochastic optimization and CCP. The problem and system model are defined in Section III. In Section IV, the stochastic formulation and deterministic equivalent are proposed. The performance evaluation of the proposed framework is presented and discussed in Section V. Finally, we conclude our findings in Section VI and present insights into future directions

II. Related Work and Background

A. NDN Mobility Management

The communication of NDN is receiver-driven where users send *interests*, i.e. *data* requests, to content providers. We will refer to the requester as the *Consumer* and the *data* owner as the *Producer*. Each node in NDN has two basic structures to perform *interests* and *data* forwarding. The Forwarding Information Base (FIB), with similar functionality to the IP table, is used to forward *interests* to *producers*. The second structure is the Pending Interest Table (PIT), which keeps track of unsatisfied *interests* as breadcrumbs to route back *data* to *consumers*. Caching is one of the main functionalities of NDN, the Content Store (CS) is available in every NDN node to support this feature. Seamless mobility support is a requirement in the future Internet since mobile events are inevitable nowadays. However, NDN supports mobility by retransmitting *interests* during routing update time, which is not scalable and can not be used to support seamless operations. Therefore, multiple schemes with different approaches have been proposed in the literature to solve mobility problems.

We classify the approaches in the literature to non-predictive and predictive mobility management schemes. The first class includes schemes that take actions after the handover event. For instance, Mobility anchor schemes [7]–[9] are based on MobileIP protocol used in the current Internet. In particular, this protocol uses special nodes in every home network called anchors that forward the *interests* to the *producer* even if the latter is moving to a new network (i.e., roaming). Another non-predictive approach is Location Resolution schemes [10]–[13] which are similar to the one used in Domain Name Systems (DNSs), where the *consumer* queries the location of the *producer* before sending *interests*.

The second class includes proactive schemes that use prediction to support mobility as in [6], [14]–[16]. By utilizing the cache resources in the network, these scheme decide on the *data* to be stored before handover which avoids *interests* droppings and retransmissions. However, the schemes assumed idealistic conditions with complete knowledge of the future *interests* in order to define the maximum prediction gains. In this paper, we propose a *robust* framework that handles uncertainties in predictions. Such uncertainties might result in ignoring some future content that increases the delay and retransmissions, while storing unnecessary *data* that increases the network overhead and lead to suboptimal utilization of caching resources.

B. Stochastic Optimization

Robust optimization refers to a class of optimization theory in which the formulated programming model contains uncertain variables [17]. In principle, two schemes are used namely *fuzzy* and *stochastic* optimization. The former models the uncertain information as fuzzy numbers with membership functions to capture the variations. The *stochastic*, on the contrary, models the uncertain information as random variables whose Probability Density Function (PDF) or PMF is used to represent the possible outcomes of prediction. Here, we focus on *stochastic* optimization which handles the uncertainty in constraints by CCP technique. In essence, CCP transforms the constraint with random variables into a *probabilistic* model that ensures its satisfaction by a minimum degree denoted by $\beta \in [0, 1]$.

In order to obtain a closed form solution, a deterministic equivalent is typically derived based on the statistical model of the random variable. The deterministic form can be obtained by means of SA, Gaussian Approximation (GA) or *Markov Inequality*, among others [17]–[19]. In this framework, we adopt the SA which uses the PMF of random variable to construct possible scenarios of the optimization model. The resultant model guarantees that the solution of such approximation will satisfy the scenarios with a probability sum of β. This is in addition to its capability of obtaining an ILP model that does not change the order of magnitude of the problem at hand

III. System Model and Problem Definition

A. System Model

Assume U is the set of users connected to the network. Each user $u \in U$ can be either a *consumer* or a *producer* for *data* $d \in D$, where D is the list of all *data* items in the network. The network topology is represented by a graph $G = (V, E)$, where each node in V is either a user or a router. Each router r in the set of routers R has a current cache size denoted by

c_r and a maximum cache capacity of c_r^{max}. Let the path taken by an *interest* or *data* from node x to node y is denoted by a sequence of nodes $P_{x\to y}$. Accordingly, the cardinality $|P_{x\to y}|$ represents the number of hops traversed using this path.

B. Handling ICN Producer Mobility

Due to the mobility of users, the *producer* is not reachable after handover until the network updates the routing tables with the new location. As a consequence, *interests* issued by *consumers* during this period will be dropped by the network. This affects the *consumer*'s experience which will suffer from long delays in addition to *interests* retransmissions which increase the network overhead. The main goal of the proposed scheme is to avoid *interests* drops which can be achieved by satisfying *interests* without the need of reaching the *producer* at his new location. The scheme accomplishes this by utilizing network resources, caches in particular, to store the *data* needed before the *producer*'s handover.

The proactive approach requires a knowledge of the future to operate seamlessly. In particular, the scheme needs to detect the *producer* mobility event before it occurs in addition to the future requests that will be sent by *consumers* to such mobile *producer* during handover. We remark, mobility and content, are highly predictable due to patterns of user movements and content requests.

1) Mobility Prediction: Location predictors as in [20], [21] can be used to predict *producer* mobility. These tools adopt location history, current position and trajectories to predict the future user locations in the network. Using the estimated location, the knowledge of network topology and user access rights, the user current and future Point of Attachment (PoA)s and the nearest caching routers can be determined. Since the scheme acts proactively, before the *producer* actually moves, the future PoA is not required to be predicted. In particular, the scheme needs to know whether the *producers* will change its PoA in the near future or not. Our prior work showed that the performance of such proactive scheme is not sensitive to the mobility estimation errors [6] due to the needed coarse level of prediction.

2) Data Prediction and Uncertainty Model: The second information is the *consumers* requests pattern which can be predicted using various techniques such as reservoir computing, stochastic models of user behavior and biology-inspired survival analysis [22], [23]. Users preferences, historical request patterns and early popularity measures are used to predict future requests. These tools are currently used in the Internet in different applications such as YouTube video recommendations and Facebook news feed.

Both future mobility and *data* are used to predict the *interests* that will to be dropped due to *producer* handover. Given the set of *interests*, we find the optimal *data* placement on in-network caches such that generated *interests* get satisfied with minimal overhead and bounded delay. Since predictors do not generate information with perfect accuracy, the estimated set of *interests* can be erroneous. Therefore, we use stochastic optimization to provide a robust optimal solution to the *data* placement problem given the possible realizations of each *interest* from the predictor. Each *data* d is represented as a random variable $\tilde{d}$ to model the prediction uncertainty. Each realization of this random variable, e.g. represents a video content encoded in different quality, is denoted by $s \in S_d$ and has probability $\pi^{d,s}$.

IV. Problem Formulation

A. Stochastic Formulation

We formulate the *data* placement problem using CCP in which the objective is to minimize the overhead while the delay is bounded by a probabilistic level under limited cache capacity. The decision variables determine where to place the *data* and which *data* to remove from the caches. The first decision variable is denoted by $\delta_r^{\tilde{d}}$ which equals 1 if the erroneous predicted *data* $\tilde{d}$ will be stored in cache r, and equals 0 otherwise. The second decision variable is ρ_r^m, which decides on how many items should be removed from the cache to provide a space for the new *data*. In particular, if ρ_r^m is 1, then m items will be removed from router r. The complete formulation is depicted as:

$$\min_{\delta_r^{\tilde{d}},\rho_r^m} \sum_{\tilde{d}\in D}\sum_{r\in R} \mathbb{P}^r_{u(\tilde{d}),u'(\tilde{d})}\delta_r^{\tilde{d}} \tag{O}$$

Subject to:

$$\sum_{r\in R}\delta_r^{\tilde{d}} \le 1, \quad \forall \tilde{d}\in D \tag{C1}$$

$$Pr\left\{\sum_{r\in R}|P_{u(\tilde{d})\to r}|\delta_r^{\tilde{d}} \le \zeta\right\} \ge \ \beta,\ \forall \tilde{d}\in D \tag{C2}$$

$$\sum_{m=0}^{c_r}\rho_r^m = 1, \quad \forall r \in R \tag{C3}$$

$$c_r + \sum_{\tilde{d}\in D}\delta_r^{\tilde{d}} - \sum_{m=0}^{c_r} m\rho_r^m \le c_r^{max}, \quad \forall r\in R \tag{C4}$$

$$\sum_{\tilde{d}\in D}\delta_r^{\tilde{d}} - \sum_{m=0}^{c_r} m\rho_r^m \ge 0, \quad \forall r\in R \tag{C5}$$

$$\sum_{m=0}^{c_r} m\rho_r^m < \alpha, \quad \forall r\in R \tag{C6}$$

The objective function in Equation (O) minimizes the amount of overhead generated by the scheme. In particular, it minimizes the total summation of path updates messages to cache the required *data* in the network. The number of path updates for each *data* d is calculated as $\mathbb{P}^r_{u(d),u'(d)} = (|P_{u(d)\to r}| - |P_{u(d)\to u'(d)} \cap P_{u(d)\to r}|)$ which represents the number of non-common edges between $P_{u(d)\to u'(d)}$ (path from the *consumer* $u(d)$ to the *producer* $u'(d)$) and $P_{d(u)\to r}$ (new path from the *consumer* $u(d)$ to the chosen router r).

The constraints (C1) to (C6) are of 3 types to; 1) cap the *consumer* delay, 2) avoid violating caching capacities, and 3) govern the range of the decision variables. In particular, Constraint (C1) ensures that *data* are cached no more than once in the network. Caching Constraints (C3) to (C6) are related to capacity and replacement. Exceeding the max capacity of any router is avoided by Constraint (C4). To avoid unnecessary replacements in each router, constraint (C5) is used. The number of replacement per router is bounded by a threshold α in (C6). The probabilistic constraint in (C2) ensures that the network caches enough content for each predicted *data* such that the delay requirement ζ is satisfied by a minimum level β. Indeed, the above formulation does not have a closed form solution due to the random variable and probabilistic constraint. Thus, we propose the deterministic equivalent model below.

B. Deterministic Equivalent

The stochastic formulation will be replaced by the following deterministic equivalent using SA. In essence, the SA will adopt the PMF of the *data* realizations to replace the random variable $\tilde{d}$. Herein, the first decision variable will thus become $\delta_r^{d,s}$ which equals 1 if the optimizer decides to cache realization s of *data* d in router r, and equals 0 otherwise.

$$\min_{\delta_r^{d,s},\rho_r^m} \sum_{d\in D}\sum_{s\in S_d}\sum_{r\in R}\mathbb{P}^r_{u(d,s),u'(d,s)}\delta_r^{d,s} \tag{O$'$}$$

Subject to:

$$\sum_{r\in R}\delta_r^{d,s} \le 1, \quad \forall d\in D, \forall s\in S \tag{C1$'$}$$

$$\sum_{r\in R}|P_{u(d,s)\to r}|\delta_r^{d,s} \le \zeta \quad \forall d\in D, \forall s\in S_d \tag{C2$'$}$$

$$\sum_{s\in S_d}\sum_{r\in R}\delta_r^{d,s}\pi^{d,s} \ge \beta \quad \forall d\in D \tag{C2$''$}$$

$$c_r + \sum_{d\in D}\sum_{s\in S_d}\delta_r^{d,s} - \sum_{m=0}^{c_r} m\rho_r^m \le c_r^{max}, \quad \forall r\in R \tag{C4$'$}$$

$$\sum_{d\in D}\sum_{s\in S_d}\delta_r^{d,s} - \sum_{m=0}^{c_r} m\rho_r^m \ge 0, \quad \forall r\in R \tag{C5$'$}$$

C3, C6

The constraints in (C2$'$) and (C2$''$) are used to replace the probabilistic constraint (C2). In essence, (C2$'$) selects the realizations to be cached for each *data* such that the delay constraint is satisfied. This is complemented by (C2$''$) which ensures that the total probability of cached realizations, for each *data*, surpasses the minimal level β. Moreover, the new objective function (O$'$) and constraints in (C4$'$) and (C5$'$) are replacing the corresponding stochastic equations by using the *data* realizations (d, s) instead of the random variable $\tilde{d}$.

In the above formulation, the value of β provides a trade-off between the risk of violating the delay requirement and the network gain quantified by the overhead in the objective function. A high value of β will force the network to cache more *data* realizations in order to increase the likelihood of satisfying the delay requirement under erroneous prediction. However, such greedy caching results in high overhead which compromises the network gains. On the other hand, choosing a low value for β will result in caching less realizations to minimize the objective function. However, this strategy increases the risk of missing the actual future *data*, resulting in high chance of violating the delay constraint (C2$'$).

The objective function and all the constraints are now linear. Moreover, the values of the decision variables are either 0s or 1s. Hence, the optimization problem is a 0-1 ILP which can be solved by numerical optimization techniques.

TABLE I
SIMULATION PARAMETERS

	Parameter	Value
General	Simulation Duration	1000s
	Transit Period	80s
	Map size	$1400m \times 1400m$
	Number of Blocks	7×7
	Number of Users	100
	Producers	50
	Consumers	50
Application	Interest Rate	50-80 *I*/s
	Zipf's s	0.2
	Content per *Producer*	$1000 \times 1KB$
Topology	APs	49
	AP Range	200m
	Number of Routers	40
	Core router's links	10Mbps
	Access router's links	5Mbps
	Propagation delay	10ms
Mobility	Model	Manhattan
	Handover delay	0.5s
	Speed	70 km/h
NDN	Forwarding Scheme	BestRoute
	Cache replacement	LRU
	Cache Size	2000 objects

V. PERFORMANCE EVALUATION

Using the assessment framework in [24], we evaluate our proposal under different scenarios.

A. Experimental Setup

The experiment used for the performance evaluation consists of 50 *consumers* and 50 *producers* in a 7x7 street grid plan with an access point in every intersection. *Consumers* request *data* with a rate of 50 to 80 *interests* per second. Moreover, the popularity of the requested *data* is based on Zipf distribution with $s = 0.2$. The network topology in the experiments is a Tran-Stub topology with 5 domains and 40 core routers. Every core router has a content store that can cache 1000 items and uses Least Recently Used (LRU) replacement policy. The parameters of the experiment are summarized in Table I.

Prediction uncertainty is simulated in the experiment design by adding errors to *interests* names. In particular, a set of *interests* is randomly chosen to be erroneous. The size of the set is based on the factor *percentage of error*, where 0% error percentage means no *interest* is erroneous and 100% means all *interests* are wrongly predicted. Then, the names of these *interests* are changed, according to the zipf distribution, to a new random (but valid) name.

B. Schemes and Evaluation Metrics

We evaluate the proposed *stochastic* scheme and compare it with: 1) the *non-proactive* Mobility Anchor (*MA*) scheme [7], and 2) the *non-robust* proactive benchmark in [6] which caches the most probable *data* realization. The following performance metrics are considered in the evaluation to reflect both the *consumer* satisfaction and the network gains: 1) *Consumer* Delay: is the total *data* access time, and is calculated as the time difference between the first attempt of sending the *interest* and successfully receiving the *data*. It includes the total timeout period and the time taken to retransmit *interests* and receive *data* packets. 2) Retransmissions ratio: is the ratio of the total number of retransmitted *interests* to the total

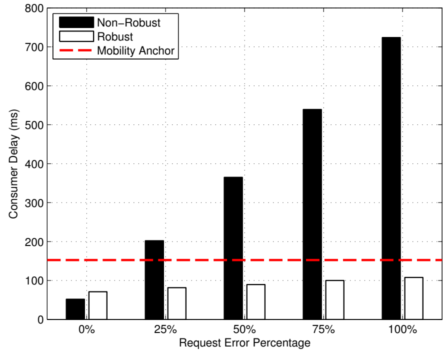

Fig. 1. The average *consumer* delay with different error percentages.

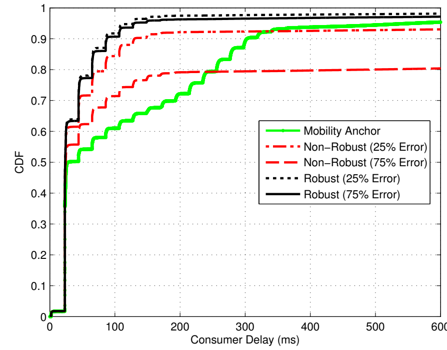

Fig. 2. The CDF of *consumer* delay with 25% and 75% error percentages.

interests sent by *Consumers*. 3) Overhead: calculated as the percentage of the total number of control packets generated by the scheme to the total number of *data* delivered.

C. Robustness Gains and Costs

We discuss the impact of errors on the schemes, where the probabilistic level β is set to a relatively high value of 0.95 that quantifies the performance bounds of adopting a *robust* scheme. Figure 1 depicts the average *consumer* delay for different error percentages. In the idealistic scenario, without prediction errors, both *non-robust* and *robust* benchmarks have a lower delay than the *MA* scheme. Specifically, the delay is reduced by 65% and 53% for the *non-robust* and *robust* compared to the *MA* scheme. This gap in performance shows the benefits of proactive caching in the network prior to mobility events.

When errors are introduced to the *data* prediction, the *non-robust* scheme fails to maintain the performance gains. Hence, the delay is increased by 30% and 377% at 25% and 100% error percentages, respectively. Assuming idealistic scenarios, by the *non-robust* scheme, resulted in caching only the most probable *data* while ignoring other realizations. The actual requested *data*, which are not cached, will be sent after the *producer* moves to the new location and sends path updates to the network. Hence, caused more delay than the *non-proactive MA* scheme which forwards the *interest*, after handover, to the *producer* in the new network. On the contrary, the proposed *robust* scheme cached more realizations to maintain the *consumer* delay. Hence, it achieves a delay based *prediction gain* of 28% in the worst case scenario (i.e. 100% prediction error) when compared to the *MA* scheme.

The Cumulative Density Functions (CDFs) of the *consumer* delay using the three approaches are depicted in Fig.2 with 25% and 75% prediction request error. While the *non-robust* outperformed the *MA* in case of low prediction error (i.e. 25%), a higher error of 75% resulted in a non-tolerable maximum delay (more than $600ms$) by the former in 20% of the cases. Unlike the *non-robust* and *MA* schemes, the *robust* scheme achieved a stable performance with a 90^{th} percentile of *consumer* delay below $100ms$. This is in addition to capping the maximum gain below the *MA* scheme in all cases.

The above gains of robustness come at the cost of increased overhead due to the control packets needed for proactive caching of *data*. The overhead of the three schemes is summarized in Table II for different mobility percentages (i.e. 25% to 75% mobile *producers*). In case of the *robust* scheme, the overhead varies between 2.03% to 2.09% which is lower than the *MA* scheme. In essence, *MA* sends large number of control packets for every handover event, while the *robust* scheme sends aggregated control packets every optimization window, see Section III. However, the *non-robust* overhead is less than the *robust* scheme by 17%. These extra control packets sent by the *robust* scheme, referred to as the cost of robustness, are due to caching more *data* to handle the uncertainties of prediction. While the *robust* scheme provides a benchmark for the delay, the *non-robust* scheme is still used to identify the minimum overhead.

However, the increase of overhead saves 83% of *interests* retransmissions, where the retransmission ratios in 100% error percentage are 0.85 and 0.15 for *non-robust* and *robust* schemes respectively. This metric is an indication of how effective is the *robust* scheme in avoiding *interests* droppings to guarantee optimal utilization of network resources. Moreover, the ratio in the *non-robust* scheme is enormously increased from 0.01 to 0.85 because of failing to cache the accurate *data*. On the other hand, the *robust* scheme has 57% fewer retransmissions than the *non-proactive* MA scheme. This is due to the reactive approach taken by *MA* which leads to retransmissions when the *producer* is in the registration phase.

D. Probabilistic Level Design Trade-off

This experiment is designed to evaluate the effect of the probabilistic threshold β on both the costs and gains of the *robust* scheme. In particular, the scheme is tested under 50% of prediction error and using β ranging from 0.7 to 1. As depicted in Fig. 3, the delay experienced by the *consumer*, i.e. robustness gain, gets shorter when β is increased. The *MA* scheme has a delay of $152.5ms$ which is an upper delay bound to the *robust* scheme for a fixed prediction error and mobility percentages. On the other hand, the idealistic scenario is simulated by using the *non-robust* scheme with no prediction errors, thus provides the lower bound for the delay (i.e. maximum prediction gains) which is found to be $52ms$.

With respect to the overhead, 3 different values for β are illustrated in Table II. As shown, the overhead increases with the selected value of β due to caching more *data* to satisfy

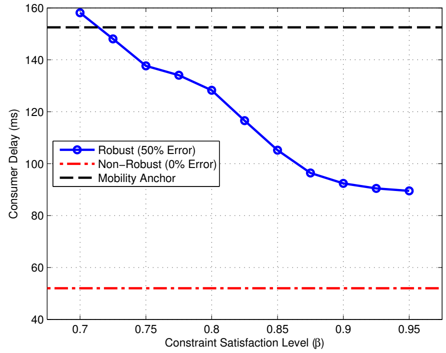

Fig. 3. The average *consumer* delay with different values of β.

TABLE II
THE OVERHEAD PERCENTAGE OF ALL SCHEMES

	Producer Mobility		
Scheme	25%	50%	75%
Mobility Anchor	2.24%	2.39%	2.40%
Non-Robust	1.74%	1.74%	1.64%
Robust, $\beta = 0.95$	2.09%	2.08%	2.03%
Robust, $\beta = 0.85$	2.04%	2.00%	1.94%
Robust, $\beta = 0.75$	1.98%	1.97%	1.88%

the delay constraints C2$'$ and C2$''$. However, such increase is negligible and maintains the superiority of the proactive *robust* scheme over the non-proactive *MA*. In essence, the value of probabilistic level β allows network designers to control the trade-off between robustness gains, represented by the *consumer* delay in our case, and the robustness costs represented by network overhead.

VI. Conclusions and Future Work

Managing mobility in NDN is a primary requirement for seamless network operations under both *consumer*'s and *producer*'s mobility. This work proposed a robust optimal *producer* mobility management scheme that exploits in-network resources and data prediction. With different level of uncertainties in prediction, the results demonstrated the ability of the proposed scheme to proactively cache future data and avoid *interests* dropping. By generating 2% of control packets, the scheme is successful to maintain lower delays compared to the *non-robust* scheme primarily proposed in the literature under idealistic prediction. Moreover, the number of *interests* retransmissions is minimal since the *interests* are satisfied before any droppings. Nevertheless, performance gap between the existing mobility management solutions and the *proactive* schemes can be controlled by the probabilistic level defined in our proposed *robust* scheme. Such outcomes also provide new trade-off between the cost of increasing the overhead and the risk of violating the *consumer* delay in the light of the prediction uncertainty.

Our future work will consider the applicability of the predictive mobility management scheme in practice. This includes studying the computational complexity and provide guided robust heuristic techniques instead of optimal solvers. Hence, provides real-time near-optimal caching decisions under prediction uncertainties.

References

[1] "Cisco visual networking index: Forecast and methodology, 2014–2019," White Paper, Cisco, May 2015.

[2] L. Zhang, A. Afanasyev, J. Burke, V. Jacobson, K. Claffy, P. Crowley, C. Papadopoulos, L. Wang, and B. Zhang, "Named data networking," *SIGCOMM Computing Commun. Rev.*, vol. 44, no. 3, pp. 66–73, Jul 2014.

[3] G. M. Brito, P. B. Velloso, and I. M. Moraes, *Information Centric Networks: A New Paradigm for the Internet*. Wiley, 2013.

[4] Z. Yan, J.-H. Lee, Y.-J. Park, and X. Lee, "Network mobility support in content-centric internet," *Int. J. of Elect., Comput., Energetic, Electron. and Commun. Eng.*, vol. 10, no. 1, pp. 91 – 97, 2016.

[5] H. Farahat and H. S. Hassanein, "Supporting consumer mobility using proactive caching in named data networks," in *IEEE Global Commun. Conf. (GLOBECOM)*, Dec 2016, pp. 1–6.

[6] H. Farahat and H. Hassanein, "Optimal caching for producer mobility support in named data networks," in *IEEE Int. Conf. Commun. (ICC)*, May 2016, pp. 1–6.

[7] J. Lee, S. Cho, and D. Kim, "Device mobility management in content-centric networking," *IEEE Commun. Mag.*, vol. 50, no. 12, pp. 28–34, 2012.

[8] F. Hermans, E. Ngai, and P. Gunningberg, "Global source mobility in the content-centric networking architecture," in *Proc. ACM Workshop Emerging Name-Oriented Mobile Networking Design - Architecture, Algorithms, and Appl.*, 2012, pp. 13–18.

[9] D. Han, M. Lee, K. Cho, T. Kwon, and Y. Choi, "Publisher mobility support in content centric networks," in *Int. Conf. Inform. Networking (ICOIN)*, Feb 2014, pp. 214–219.

[10] D.-h. Kim, J.-h. Kim, Y.-s. Kim, H.-s. Yoon, and I. Yeom, "End-to-end mobility support in content centric networks," *Int. J. Commun. Syst.*, vol. 28, no. 6, pp. 1151–1167, 2015.

[11] Z. Zhu, A. Afanasyeva, and L. Zhang, "A new perspective on mobility support," NDN, Technical Report NDN-0013, 2013. [Online]. Available: http://named-data.net/publications/techreports/trmobility/

[12] A. Azgin, R. Ravindran, and G. Wang, "A scalable mobility-centric architecture for named data networking," *CoRR*, vol. abs/1406.7049, 2014. [Online]. Available: http://arxiv.org/abs/1406.7049

[13] Z. Zhou, X. Tan, H. Li, Z. Zhao, and D. Ma, "MobiNDN: A mobility support architecture for NDN," in *Chin. Control Conf. (CCC)*, Jul. 2014, pp. 5515–5520.

[14] T. Woo, H. Park, S. Jung, and T. Kwon, "Proactive neighbor pushing for enhancing provider mobility support in content-centric networking," in *Int. Conf. Ubiquitous and Future Networks (ICUFN)*, Jul. 2014, pp. 158–163.

[15] M. Lehmann, M. Barcellos, and A. U. Mauthe, "Providing producer mobility support in ndn through proactive data replication," in *Proc. of IEEE/IFIP Network Operations and Manage. Symp. (NOMS)*, 2016.

[16] H. Farahat and H. S. Hassanein, "Proactive caching for producer mobility management in named data networks," in *Int. Wireless Commun. and Mobile Computing Conf.*, June 2017, pp. 171–176.

[17] P. Kali and S. W. Wallace, *Stochastic Programming*. Springer, 1994.

[18] R. Atawia, H. S. Hassanein, and A. Noureldin, "Fair robust predictive resource allocation for video streaming under rate uncertainties," in *IEEE Global Commun. Conf. (GLOBECOM)*. IEEE, 2016, pp. 1–6.

[19] R. Atawia, H. S. Hassanein, H. Abou-Zeid, and A. Noureldin, "Robust content delivery and uncertainty tracking in predictive wireless networks," *IEEE Trans. Wireless Commun.*, vol. 16, no. 4, pp. 2327–2339, 2017.

[20] N. Samaan and A. Karmouch, "A user centric mobility prediction approach based on spatial conceptual maps," in *IEEE Int. Conf. Commun. (ICC)*, vol. 2, May 2005, pp. 1413–1417.

[21] K. Laasonen, "Route prediction from cellular data," in *Proc. 2nd Workshop Context Awareness for Proactive Syst.*, 2005, pp. 147–158.

[22] X. Chen and X. Zhang, "A popularity-based prediction model for web prefetching," *IEEE Computer*, vol. 36, no. 3, pp. 63–70, Mar 2003.

[23] J. G. Cleary and I. Witten, "Data compression using adaptive coding and partial string matching," *IEEE Trans. Commun.*, vol. 32, no. 4, pp. 396–402, Apr 1984.

[24] H. Farahat and H. Hassanein, "On the design and evaluation of producer mobility management schemes in named data networks," in *ACM Int. Conf. Modeling, Anal. and Simulation Wireless and Mobile Systems(MSWIM)*, 2015, pp. 171–178.