A proactive multi-level cache selection scheme to enhance consumer mobility support in Named Data Networking

Lanlan Rui, Suijia Yang and Haoqiu Huang

Abstract
Resending Interest packets to support consumer mobility in Named Data Networking usually leads to latency, which can severely affect the performance of applications with strict latency requirements. We propose a proactive multi-level cache selection scheme to enhance the consumer mobility support in Named Data Networking. This scheme focuses on a handover situation, wherein the consumer disconnects from a network or moves to another access router before receiving the requested content. The proactive multi-level cache selection is designed to prefetch contents into a subset of the neighboring routers, to achieve a trade-off between the latency and cache cost. In particular, this approach is feasible for a two-level router structure and scalable to a multi-level structure. The results obtained from simulation show that compared with equivalent schemes such as the original no-prefetch scheme in the Named Data Networking and the all-prefetch scheme, proactive multi-level cache selection for a two-level router structure can provide a better handover performance to some extent; the user experience is improved and the negative impacts of consumer mobility are alleviated.

Keywords
Named Data Networking, consumer mobility, proactive cache, neighbor selection, multi-level network structure

Introduction
Information-centric networking (ICN) is a new network architecture different from the IP network architecture,1 which has problems such as inefficiency and limited scalability, due to the immense changes in the Internet environment and applications in recent years. It has emerged as a promising candidate for the architecture of the Future Internet.2 There are many projects under the ICN architecture3 such as the Data-Oriented Network Architecture (DONA),4 Publish-Subscribe Internet Routing Paradigm (PSIRP),5 Network of Information (NetInf) from the Design for the Future Internet (4WARD) project,6 and the focus of this article—Named Data Networking (NDN), synonymous with content centric networking (CCN).7

NDN shares significant aspects of the ICN architecture, such as assigning names to content instead of assigning IP addresses to hosts, adopting the Publish/Subcribe communication model, and implementing a general in-network cache. Specifically, it replaces IP addresses with content names to dissociate the content from its storage location, so that the operations of content consumers and producers are independent in time.
and space. A consumer drives every content transmission process followed by a response from a producer. A generic in-network cache could reduce the latency of requests for popular content. Furthermore, there are only two types of packets in NDN—Interest packets sent out by consumers to request content and Data packets returned from content producers or caches in NDN routers. The Interest packets contain content names and the Data packets carry the content corresponding to specific Interest packets.8

However, NDN still faces many challenges. One of the challenges is the mobility support for both the content consumer and producer. In this article, we study the mobility support for the consumer.

To simplify our analysis, we consider all the network nodes in NDN as routers; in addition, routers at the edge are considered to be wireless access points. According to the design of the generic in-network cache, these routers will cache the Data packets delivered through them. When roaming around, the consumer may frequently change the router with which it directly communicates, due to the variation in physical connectivity (or signal strength). This process is called a handover. Sometimes, a consumer starts a handover before receiving all the content that has been already requested via Interest packets. An example of this case is illustrated in Figure 1.

A content consumer is initially at location 1 and requests content from Router $i$. Subsequently, the consumer moves to location 2 resulting in a disconnection from Router $i$, but the content receiving process has not finished. Then the consumer connects to Router $f$. This is a complete process of the handover. The consumer should resend Interest packets to Router $f$ to request the content that has already been requested but not received from Router $i$. This original NDN mechanism can support the consumer mobility to some extent. Nevertheless, since the Data packets stored in the routers along the track (from the producer to Router $i$) are not accessible until the completion of update of relevant Forwarding Interest Base (FIB) entries, retransmitted Interest packets will be forwarded hop-by-hop toward further routers, including the original producer. As a result, such a handover would waste resources spent on retransmissions and lead to significant latency, which cannot be ignored for applications with strict latency requirements, such as teleconferencing, online games, and real-time notification services.

The proactive multi-level cache selection (PMCS) scheme proposed in this article aims at optimizing the aforementioned original NDN scheme for consumer mobility support. Special packets similar to Interest packets are designed to notify routers of handover information. Potential new routers (i.e. the ones the consumer may connect to after a handover) near the current router (called C-router for short, i.e. the one the consumer will disconnect from during the handover) will be selected to cache the desired Data packets in advance. The goal of PMCS is to reduce the latency and packet loss, accomplish a high average cache hit ratio, and limit the overhead of cache or signal to an acceptable scale during the handover process.

The rest of the article is organized as follows. Related work is summarized in section “Related work.”

![Figure 1](image-url)  
*Figure 1. A typical handover scenario: a consumer moves before receiving requested content.*
In section “The formulation of PMCS,” we introduce the PMCS scheme in detail by introducing the scheme model, that is, the neighbor router selecting strategy. In section “Performance evaluation,” we present the performance evaluation. The conclusion and further work of this study is highlighted in section “Conclusion.”

**Related work**

Mobility support has been widely studied in traditional networks, but the existing solutions cannot be extended to NDN. Currently, there are some special schemes for NDN. Kim et al.9 introduced the existing problems in mobility scenarios of NDN but proposed no effective solutions. In reactive schemes such as the one by Caporuscio et al.10 proposed for traditional networks, routers communicate with proxies. Then the proxy caches items for the consumer during the handover. When reconnecting to the network, the consumer informs its new router of the identity of the previous proxy; subsequently, the new router sends Interests to the previous proxy in order to request all the items that have been cached during the disconnection period. This reactive procedure has the drawback of increased delay, since the new router must receive all the cached items before it can start to forward them. Lee and colleagues11,12 have designed a scheme utilizing in-network cache to reduce average latency, but this scheme still relies on the IP network. Rao et al.13 presented a proactive cache scheme by exploiting the routers in CCN. The previous router in this scheme forwards content to the new router when the handover starts; however, the new router has to be confirmed first, which is impractical in some situations, for example, fast moving states and emergency environments with an unstable network status. Vasilakos et al.14 propose a proactive selective neighbor caching scheme. It balances the latency and cache cost to choose an optimal set of neighbor cache servers (one hop away) to store the content. This scheme is similar to ours in some aspects. However, there are several significant different features. The PMCS considers the structure of the router—a multi-level structure as well as a flat structure; furthermore, it improves the definition of neighbor routers and sets an upper bound to the cache size to avoid excessive resource consumption.

**The formulation of PMCS**

In this section, we describe the scheme we designed to improve the consumer mobility support in NDN. The scheme overview will be introduced first, followed by designs of the packet format, whole the flow, and router selection algorithms.

**The scheme overview**

Figure 2 shows the overview of the proposed method. An NDN consumer in motion detects changing link information or signal strength to judge whether a handover is going to happen. If a handover is due soon, the consumer will notify the router to which it is connected. Subsequently, the router will select a subset of routers from neighbor routers (defined in section “Neighbor routers and transition probability”) to receive the content that has already been requested. The consumer stops requesting content once the handover actually starts. After reconnecting to a new router, the consumer wants to receive content immediately instead of waiting for the new router to multicast Interests. Simultaneously, routers that have prefetched content will stop the corresponding cache process, if they have not already stopped due to the absence of desired content or limit of cache size. Specific designs necessary for this whole process will be articulated in this section.

**Names, packets, and the workflow**

Hierarchical structure names are allocated to edge routers, which the consumer could directly connect to. Any edge router’s complementary name is added into FIBs of the routers within the same domain. Considering the scalability, the names of routers beyond the domain will be added only to the gateway routers. Special packets based on the original Interest packet are designed as shown in Figure 3. In Figure 3(a), an optional field type, added to the Interest format,
represents the type flag of the special packets. Figure 3(b) shows types, flags, and name fields in detail.

To simplify the description of our scheme, we define the following terms in this article: the previous router (P-router) represents the router from which the consumer disconnects during a handover; the new router (N-router) represents the router to which the consumer connects after a handover; and a selected (S-router) is a router selected by the P-router to prefetch content for the consumer. There may be more than one S-router and S-routers may include the N-router.

The workflow of the Data and Interest packets among the various routers is illustrated in Figure 4. The normal steps before and after the handover process are the same as their counterparts in the original NDN, so they are not indicated. Furthermore, since the producer is stationary, we do not indicate the router connected to the producer:
Step 1. The P-router recognizes an Out packet from the type flag and recognizes that a handover is imminent.
Step 2. The P-router assigns a time stamp about the imminent handover and selects a subset of routers as S-routers (the selection algorithm is introduced in section “The selection algorithm”). In an ideal case, the N-router is one of the S-routers. A Neighbor packet, carrying the names of S-routers and the time stamp, will be sent to the consumer.
Step 3. The P-router sends Cache packets, each carrying a name of the content that the consumer has requested via an Interest packet but not yet received, to S-routers.
Step 4, 5. S-routers start to prefetch contents from the producer.
Step 6. Upon reconnecting into NDN via the N-router (i.e. the handover has finished), the consumer sends normal Interest packets and Neighbor packets to the N-router.
Step 7. The N-router responds to Interest packets with normal Data packets.
Step 8. The N-router informs the S-routers to stop prefetching.

The selection algorithm

Neighbor routers and transition probability. The P-router selects routers from a set of neighboring routers. Vasilakos et al.14 defined routers one hop away from the current router as neighbor routers. However, the consumer may reconnect to routers more than a hop away due to the placement of routers, signal strength, and physical distance between the routers, which often does not match the topological structure of routers. We consider neighbor routers to be the ones to which the consumer may connect after a handover. Hence, we determine the neighbor routers dynamically, similar to the AP Neighbor Graph proposed by Mishra et al.15

Define a sequence M(c) to represent the mobility pattern of consumer c

M(c) = \{(r_1, t_1, l_1^0, r_2, t_2, l_2^0), \ldots, (r_n, t_n, l_n^0)\} \tag{1}

where r_i is the router to which the consumer connects at t_i and disconnects from l_i. Hence every period from l_i to r_i+1 is a handover.

We define a graph G = (V, E) where V = \{r_1, r_2, \ldots, r_n\} is the set of all the edge routers. If there was a handover from Router i to Router j, we consider j to be a neighbor of i, and the edge e = (r_i, r_j) represents this relationship. Neighbor(r_i) = \{r_j | r_j \in V, (r_i, r_j) \in E\} represents the neighbor routers of r_i. By collecting the mobility pattern data of the consumers, individual routers can learn neighbor routers dynamically over time. Furthermore, Mishra et al.15 defined the graph as undirected, but we consider it to be directed. Because in practical situations consumers often move along a path in different patterns (e.g. with different speeds or in opposite directions). In this article, we need to not only find all the neighbor routers but also select a subset according to specific transition probability. Prefetching content into all the neighbors could ensure a high hit ratio, but the cache overhead is also high due to the increased number of neighbors. Furthermore, even if the P-router has many neighbors, at most 3 or 4 are possible targets of the handover.14

We use P_{ij} to represent the transition probability that the consumer connects to neighbor Router j after disconnecting from Router i. According to previous relevant research, such transition probabilities can be predicted or estimated by analyzing the user action data such as moving tracks16 or the ratio of the number of transitions from Router i to Router j.14 therefore, we assume that such information is available and known.

S-routers for flat structure. Prefetching content can reduce latency, but it will result in excessive cache resource consumption. Our scheme focuses on determining an optimal router subset to minimize the total cost of consumer mobility, by balancing latency cost against cache cost.

Comprehensively considering the time cost of acquiring content, cache cost of prefetching content, and corresponding weights, we define the total cost Cost_{total} of a flat router structure as

$$\text{Cost}_{total} = w_L \times L_{total} + w_C \times C_{total} \tag{2}$$

where L_{total} indicates the total latency cost with weight w_L, and C_{total} represents the total cache cost with weight w_C. Let S indicate the set of all the neighbor routers of Router i. Furthermore, S(S' \subset S) denotes the subset, which is selected to prefetch content. The probability of the consumer connecting to a Router j in S' is P_{hit}(S') = \sum_{j \in S'} P_{ij}. Since a handover usually takes place over a short period, we do not consider the possibility of cache replacement. Hence, the probability P_{hit}(S') represents the cache ratio approximately. Besides, we set an upper bound to the size of prefetching to avoid excessive cache consumption and a lower bound to the signal strength to send out packets to reduce the latency.

We denote the latency due to receiving content from a neighbor router j (j \in S') as L_{hit}' and the latency due to receiving content from the producer as L_{miss}. It is known that L_{miss} \geq L_{hit}' in usual cases.

We denote the cache cost of prefetching a content item into Router j (j \in S') as C_{cache}' and the cache cost of the producer as C_{cache}. Over a complete handover process, the whole latency L_{total} is the sum of
propose a discriminant equation as follows.

Second, we judge whether a router \( k \) should be selected into \( S \). For any router \( k \in S \), if \( P_{th} > \frac{w_j}{w_j} * \frac{C^k_{cache}}{L_{miss} - L_{hit}} \)

\[
P_{th} > \frac{w_j}{w_j} * \frac{C^k_{cache}}{L_{miss} - L_{hit}}
\]

The algorithm for selecting prefetching routers in a flat structure is specifically shown in Figure 5.

**Algorithm 1 : Selecting pre-fetching routers in flat structure**

| Step | Description |
|------|-------------|
| 1    | While the consumer moves |
| 2    | Get the current router \( i \) the user connects to; |
| 3    | Get the neighbor router set \( S \) of \( i \); |
| 4    | \( S' = \emptyset \) |
| 5    | For any router \( k \in S \) |
| 6    | If \( P_{th} > \frac{w_j}{w_j} * \frac{C^k_{cache}}{L_{miss} - L_{hit}} \) |
| 7    | \( S' = S' \cup \{k\} \) |
| 8    | \( S = S - \{k\} \) |
| 9    | ... |
| 10   | ... |
| 11   | \( S \) is the user stops moving, handover is over or pre-fetched contents exceeds the threshold. |
| 12   | \} |

**Figure 5.** Algorithm 1.

every \( L_{hit}^j \) and \( L_{miss} \), whereas the whole cache cost is the sum of every \( C^j_{cache} \) as shown in equations (3) and (4)

\[
L_{total} = \sum_{j \in S'} (P_{hit} L_{hit}^j) + (1 - P_{hit}(S')) \cdot L_{miss}
\]

\[
C_{total} = \sum_{j \in S'} C^j_{cache}
\]

The purpose of selecting prefetching routers is to minimize \( C_{total} \); hence, first, we initialize \( S' \) to be empty; second, we judge whether a router \( k (k \in S) \) should be selected into \( S' \) independently in accordance with the constraint \( C_{total}(S') > C_{total}(S' \cup \{k\}) \); third, we combine equations (2)–(4); and finally, we propose a discriminant equation as follows

\[
P_{th} > \frac{w_j}{w_j} * \frac{C^k_{cache}}{L_{miss} - L_{hit}}
\]

gateway routers. The number of levels can be more than two. Figure 6 shows an example of a three-level router structure. In this article, we mainly propose optimal strategies for a flat structure and a two-level structure. Our scheme can also be used in a multi-level router structure using a similar improvement process; hence, we do not describe it specifically in this article.

We define \( \nabla S' \) as a set modified from \( S' \). Every element in \( \nabla S' \) is a parent router of a few elements in \( S' \) and any duplicate elements are removed. Considering the example in Figure 6, \( S' = \{w, e, r\} \), \( \nabla S' = \{u, v\} \); furthermore, we consider that the routers in \( S' \) are divided into several subsets \( S'_i \) according to their parent routers. The number of elements in \( \nabla S'_i \) is denoted as \( |\nabla S'_i| \). From the above-mentioned definition, we can derive \( S'_1 = \{w, q\} \) and \( S'_2 = \{e, r, t, y\} \).

Within a two-level structure, we denote the latency due to a consumer requesting and receiving a content item from Router \( j \) (in level 2) as \( L_{hit}^{j,1} \) and denote the cache cost due to a router prefetching the corresponding content as \( C^j_{cache} \). Furthermore, \( L_{hit}^k \) and \( C^k_{cache} \) in the flat structure are modified to \( L_{hit}^{j, 1} \) and \( C^j_{cache} \), respectively.

With regard to the procedure of prefetching router selection, first, we determine \( S' \) in level 1 according to algorithm 1. If the cost of prefetching content into \( S'_i (i = 1, 2, \ldots, |S'|) \) is more than the cost of prefetching content into the corresponding parent router, we select the parent router instead of \( S'_i \).

After modifying and improving equations (2)–(4) and the constraints, we derive discriminant (6), which can judge whether to add Router \( m \) in level 2 into \( S' \).

\[
\frac{w_j}{w_k} \left[ \sum_{j \in S'_i} P_{hit} L_{hit}^{j,1} - \left( \sum_{j \in S'_i} P_{hit} L_{hit}^{m,2} \right) \right] * \sum_{j \in S'_i} C^j_{cache} \geq C^m_{cache}
\]

(6)

The algorithm for selecting prefetching routers in a two-level structure is specifically shown in Figure 7.
Performance evaluation

Simulating methods

To evaluate the performance of the scheme proposed in this article, we have conducted simulations using MATLAB. We mainly conducted experiments for a two-level router structure, simulating the scenario wherein a consumer moves in a 12 km × 1 km area and the handover happens constantly. The simulation scenario is shown visually in Figure 8.

The small black solid dot represents the consumer and its movement is automatic (the path is continuous, and we set it to be a sin function wave). The consumer connects to and disconnects from a router and completes the router handover in accordance with our scheme. In the simulation area, there are 16 routers in level 1 represented by relatively bigger circles numbered 5–20 and four routers in level 2 represented by green circles numbered 1–4. The consumer can directly connect to any router in level 1. The connected router turns to solid black dots and subsequently returns to a circle after disconnection. Once selected into $S_0$ to pre-fetch content for the consumer, the routers will twinkle and subsequently return to circles when the prefetching stops. The producer is stationary in our scheme, so we assume its coordinate of fixed location to be $1,12$.

Simulation settings

The consumer continuously collects the distance from the P-router to determine the signal strength. Once a handover is predicted, the P-router selecting algorithm will start. Since determining the neighbor routers is based on historic consumer mobility data, the data and the corresponding transition probabilities can become reasonably stable. Hence, we used fixed data for the neighbor routers and transition probabilities. Furthermore, $L_{\text{miss}}$ and $L_{\text{hit}}$ are determined by the distances between the consumer and the corresponding routers. Furthermore, the model is partially simplified in simulation experiments by assuming that each content item is of the same size; hence, each $C_{\text{cache}}^i$ in a level is the same. Some key parameters are listed in Table 1.

Analysis of results

Comparison of different schemes. According to the parameters set in section “Simulation settings,” we have conducted simulations on our scheme and several relevant schemes including the selective neighbor caching (SNC), all-prefetching scheme, which forwards content to every neighbor router after content is
transmitted back into the previous router, and the default no-cache scheme in the original NDN. The data regarding average cache cost, latency, and total cost in a handover are illustrated in Figure 9.

The default scheme of original NDN has no cache cost because it does not prefetch content. Due to the strategy of the all-prefetching scheme, the consumer can always receive the content, which was requested before the handover, from the new router, leading to low latency and high cache cost. The latency of the SNC is lower than that of the default scheme in NDN; however, its cache cost is higher than that of the default scheme. The PMCS for a flat structure has an optimized strength threshold (distance) compared to the SNC; hence, it has a lower latency but higher cache cost than the latter scheme. The PMCS for a two-level router structure utilizes fewer routers in level 1 by replacing them with routers in level 2; hence, the cache cost decreases, and the latency increases slightly. With regard to the total cost, considering the average numerical results over many independent repetitions, the PMCS for a two-level router structure costs 22.3% less than the default no-cache scheme in the original NDN, 27.2% less than the all-cache scheme, 18.3% less than the SNC, and 4.3% less than the PMCS for a flat router structure.

The influence of router density. It has been determined through experiments that the consumer sometimes switches to a new router seamlessly at areas where the signal ranges of several routers overlap. We adjust the density of routers (i.e. router numbers) and subsequently obtain the data in Figure 10 regarding the average total cost of the PMCS for a two-level structure and the default scheme in NDN.

It is observed that with increasing number of routers, the total cost of the default scheme in NDN decreases significantly, whereas the PMCS for a two-level structure decreases gradually. The performance advantage of the PMCS reduces with increasing router density and it even costs slightly more than the default scheme in NDN ultimately, due to the probability of seamless handover becoming higher and the cache cost reducing slightly. The decreasing advantage of PMCS against schemes such as the default scheme in original NDN demonstrates that PMCS is more suitable for scenarios with a long handover time or low router density.

The effect of the amount of routers on handover time is shown in Figure 11(a). Figure 11(b) shows when the handover period changes, how cache costs of all-prefetching, SNC, and PMCS change.

When the router number is small (equivalent to low router density), the consumer has to wait for a long time after disconnecting from any router before it connects to a new router; hence, the handover period is relatively long. Both the all-prefetching and SNC schemes do not have a cache threshold; hence, the total content prefetched always increases as the handover period increases. The PMCS will stop prefetching if the handover time exceeds the threshold to control cache cost.

Influence of $w_c/w_l$. Adjusting the value of $w_c/w_l$ will influence the selection of prefetching routers. The related data are shown in Figure 12.

When the value of $w_c/w_l$ is small, the PMCS for a two-level router structure selects more routers in level 1 and fewer routers in level 2. The weight of the cache cost is small when the latency cost is significant. In such
a case, selecting more routers in level 1 can reduce the latency and total cost. When $w_c/w_l$ is set to a bigger value, that is, the cache cost is more significant than latency, selecting fewer routers in level 1 can reduce the cache cost. In particular, when the number of routers selected in level 1 is small, selecting routers in level 2 can barely reduce the cache cost; hence, the number of routers selected in level 2 reduces, too.

Conclusion

In this article, we have examined consumer mobility in NDN and presented PMCS, namely, a proactive caching router selection scheme. In particular, we have underlined the balance between the latency and cache cost to minimize the total cost. It is shown that the PMCS outperforms the all-prefetching scheme in cache cost and further outperforms the SNC and the original no-prefetch scheme in NDN in latency cost. Moreover, environment conditions and the producer’s motion state may impact on PMCS’s superiority. Further insights can be gained by investigating distinct strategies for different types of movement and complex network structures. Besides, we are going to deal with more challenges in our further research, such as revising current scheme and improve it to handle mobility support for producers.

Declaration of conflicting interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.
Funding
The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: The work presented in this article was supported by the National Natural Science Foundation of China (61302078, 61372108), 863 Program (2011AA01A102), National S&T Major Project (2011ZX 03005-004-02), and Beijing Higher Education Young Elite Teacher Project (YETP0476).

References
1. Lee J, Cho S and Kim D. Device mobility management in content-centric networking. IEEE Commun Mag 2012; 50: 28–34.
2. Xylomenos G, Ververidis CN, Siris VA, et al. A survey of information-centric networking research. IEEE Commun Surv Tut 2014; 16: 1024–1049.
3. Ahlgren B, Dannewitz C, Imbrenda C, et al. A survey of information-centric networking. IEEE Commun Mag 2012; 50: 26–36.
4. Koponen T, Chawla M, Chun BG, et al. A data-oriented (and beyond) network architecture. ACM SIGCOMM Comput Commun Rev 2007; 37: 181–192.
5. Tarkoma S. PSIRP deliverable 2.3, architecture definition, component descriptions, and requirements. PSIRP 7th FP EU-funded Project, http://www.psirp.org/ (2008, accessed 12 January 2016).
6. Ahlgren B. Second NetInf architecture description. 4WARD EU FP7 Project, http://www.4ward-project.eu/ (2010, accessed 12 January 2016).
7. Zhang L, Estrin D and Burke J. Named Data Networking (NDN) project. Technical Report, PARC, Palo Alto, CA, October 2010.
8. Bari MF, Chowdhury S, Ahmed R, et al. A survey of naming and routing in information-centric networks. IEEE Commun Mag 2012; 50: 44–53.
9. Kim DH, Kim JH, Kim YS, et al. Mobility support in content centric networks. In: Proceedings of the ACM ICN workshop, Helsinki, 13–17 August 2012, pp.13–18. New York: ACM.
10. Caporuscio M, Carzaniga A and Wolf AL. Design and evaluation of a support service for mobile, wireless publish/subscribe applications. IEEE T Software Eng 2003; 29: 1059–1071.
11. Lee J and Kim D. Proxy-assisted content sharing using content centric networking (CCN) for resource-limited mobile consumer devices. IEEE T Consum Electr 2011; 57: 477–483.
12. Lee J, Kim D, Jang MW, et al. Proxy-based mobility management scheme in mobile content centric networking (CCN) environments. In: Proceedings of the IEEE international conference on consumer electronics, Berlin, 6–8 September 2011, pp.595–596. New York: IEEE.
13. Rao Y, Zhou H, Gao D, et al. Proactive caching for enhancing user-side mobility support in Named Data Networking. In: Proceedings of the 2013 seventh international conference on innovative mobile and internet services in ubiquitous computing (IMIS), Taichung, Taiwan, 3–5 July 2013, pp.37–42. New York: IEEE.
14. Vasilakos X, Siris VA, Polyzos GC, et al. Proactive selective neighbor caching for enhancing mobility support in information-centric networks. In: Proceedings of the ACM ICN workshop, Helsinki, 13–17 August 2012, pp.61–66. New York: ACM.
15. Mishra A, Shin M and Arbaugh WA. Context caching using neighbor graphs for fast handoffs in a wireless network. In: Proceedings of the IEEE INFOCOM 2004, Hong Kong, China, 7–11 March 2004, p.361. New York: IEEE.
16. Pathirana PN, Savkin AV and Jha S. Mobility modelling and trajectory prediction for cellular networks with mobile base stations. In: Proceedings of the 4th ACM international symposium on mobile ad hoc networking and computing, Annapolis, MD, 1–3 June 2003, pp.213–221. New York: ACM.
17. Malak D and Al-Shalash M. Optimal caching for device-to-device content distribution in 5G networks. In: Proceedings of the 2014 IEEE GLOBECOM workshops (GC workshops), Austin, TX, 8–12 December 2014, pp.863–868. New York: IEEE.
18. Long X and Sikdar B. A wavelet based long range signal strength prediction in wireless networks. In: Proceedings of the 2008 IEEE international conference on communications, Beijing, China, 19–23 May 2008, pp.2043–2047. New York: IEEE.