Dynamic Traffic Awareness in Wireless Multi-Channel Data Dissemination

Chih-Lin Hu[†], Ming-Syan Chen[†] and Hao-Ping Hung[‡]
[†]Department of Electrical Engineering
[‡]Graduate Institute of Communication Engineering
National Taiwan University
Taipei, Taiwan, R.O.C.

Abstract— **Data dissemination by the use of periodic broadcast is an important scalable data delivery mechanism due to the explosion of wireless information services. Consider that the recent emergence of a multi-channel data dissemination model has attracted much research attention. In this paper, we devise an on-line multi-channel traffic awareness (MCTA) mechanism by exploiting the notion of client impatience against traffic dynamics in a multi-channel data dissemination environment. The design of MCTA performs a cyclical estimation of dynamic access frequency distribution in response to dynamic traffic changes. The experimental results show that MCTA is not only able to attain an estimated distribuion of high accuracy, but also of low complexity and robustness without performance degradation.**

I. Introduction

Previous studies addressed the *push-based data broadcast/dissemination* model to resolve the scalability by the explosion of data intensive applications and to minimize the power/energy consumption due to the restriction of mobile client devices, especially in wireless network environments which have the communication asymmetry in terms of bandwidth and information flow [1][3][9][16]. For example, the downward bandwidth is larger than the opposite one in satellite networks, radio networks and cellular networks. Also, the information flow is asymmetric in Web surfing, instant messaging systems and digital video broadcasting services, to name a few. In the setting of data broadcasting, the server periodically delivers data items to a large client populations through a shared medium. Clients listen on the broadcast channel and download data items without explicit access requests. Therefore, downloading broadcast data consumes less power than actively interacting data request/reception [11].

To efficiently disseminate data, generally, the server can use the pull, the push or the hybrid data delivery model. The mean access time, specifically from the moment at which a client requests an item till the moment that the item is downloaded completely, is the major performance measure [8]. In a push delivery, the mean access time, half a cycle time, can be lengthy due to the existence of many broadcast items. However, a pull delivery responding to data requests is not scalable under a heavy workload. Thus, a hybrid model strikes a compromise [2][6][15]. Particularly, all items are classified into the push/pull (hot/cold) mode according to their access frequencies. The server periodically pushes hot items by broadcasting and carries cold items in the pull manner.

The recent advent of the multi-channel data dissemination has attracted many research attention [12][14]. In [12], the work gives an analytical study of the multi-channel partition. The work in [14] designs the multi-channel caching scheme. In [7], the broadcast disk programs on multiple channels are dynamically generated to improve energy saving. [4] designs an adaptive balanced hybrid data delivery scheme. For the purpose of application scalability, fault tolerance or reconfiguration of adjacent cells, multiple downward channels are allocated on discontiguous spectrums. In practice, it is possible that channels cannot be coalesced into a shared channel. Hence, the multi-channel model has different bandwidth utilizations and granularities of item replacement/switch as compared to the single-channel model. Explicitly, a channel switch involves a series of item demotions/promotions into the pull/push set [12][13] as opposed to one item switch in the single-channel model [2][6][15].

Notice that a hybrid data delivery should be adaptive to sustain traffic dynamics in multi-channel model. The lack of dynamic traffic awareness will make the broadcast contents reliable. However, many works in the data broadcast research were based on static workload and access patterns in a single-channel model. Although several techniques [5][6][15][17] are used to detect traffic changes, none is designed for the multi-channel model. The probing technique is firstly proposed, but has the scalability problem; the partial probing or sampling technique is used [15]. The feedback technique appends the access statistics into pull access requests with an extra feedback representation [6]. The statistic maximum likelihood estimation method is dedicated to estimate the static schedule [17]. The SDR technique estimates the dynamic access frequency distribution in a single-channel model [5].

In this paper, we propose an on-line multi-channel traffic awareness (MCTA) mechanism which exploits the nature of client impatience [10] to periodically estimate the dynamic access frequency distribution in multiple broadcast channels. Particularly, clients have limited patience, while accessing broadcast data items. A client will submit an impatient pull request when the waiting time exceeds its patience. With the measured impatient requests of each broadcast item in a broadcast cycle time, the server reflectively estimates the access frequency distribution among all items in the broadcast channels. Note that MCTA extends the original SDR [5] technique and can be elegantly applied to the push and the hybrid data delivery models without complicating the data broadcasting system. Further, we devise MCTA with all-inclusive traffic factors in order to provide a foundation for the adaptation of broadcast scheduling, indexing and the hybrid data delivery as compared to previous one which resolves a static optimization problem in a single-channel model. The

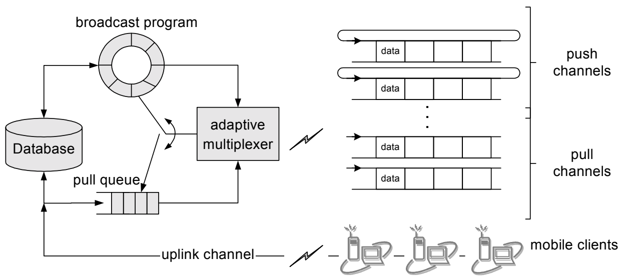


Fig. 1. A hybrid data delivery in a multi-channel data broadcast model.

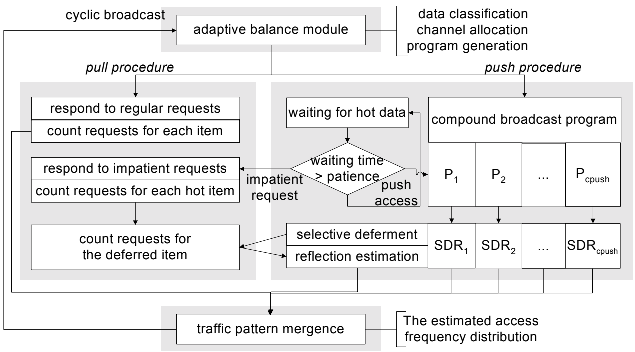


Fig. 2. The flowchart of the MCTA mechanism.

experimental results show that MCTA is of low complexity without performance degradation and is able to attain a highly accurate estimation of dynamic access frequency distribution.

The rest of the paper is organized as follows. Section II describes the design of MCTA. Section III presents the experimental results. This paper concludes with Section VI.

II. Multi-Channel Traffic Awareness (MCTA)

A. System Modeling and Notation

As depicted in Fig. 1, a wireless information server has a database $D = \{d_1, d_2, ..., d_m\}$ where items have the same size. Assume that it takes one time slot to broadcast each item, the use of time slot and data slot are interchangeable in this model. Let the request arrival rate $\lambda_i(j)$ of each item d_i in a time unit t_j form a Poisson process with an aggregate arrival rate $\lambda = \sum_{i=1}^{m} \lambda_i(j)$ where $\lambda_1(j) \geq \ldots \geq \lambda_m(j)$. All items in D are divided into a push item set $U^h = \{d_1, ..., d_k\}$ and a pull item set $U^c = \{d_{k+1}, ..., d_m\}$ according to their access frequencies (or the request arrival rates). Suppose the downward bandwidth consists of c channels of equal bandwidth. The server uses c_{push} channels to periodically broadcast items in U^h. Other c_{pull} channels carry items in U^c in the pull manner.

To maintain the generality, a *flat broadcast schedule* is used in this study. Hot items in U^h are arranged into push channels without item duplication and dummy slots. A *compound* broadcast program $P = \bigcup_{x=1}^{c_{push}} P_x = P_1 \cup \ldots \cup P_{c_{push}}$ is the union of *coordinative* broadcast programs P_xs. Specifically, $P_x = \{d_1^x, \ldots d_i^x, \ldots, d_L^x\}$ presents the broadcast program in the x-th push channel. Thus, the time interval of a broadcast cycle L is equal to the number of slots in each P_x. In addition, when a client wants to access an item d_i^x, it will submit an impatient pull request for d_i^x after the waiting time exceeds its patience w. The amount of impatient pull requests for d_i^x is denoted as γ_i^x in each broadcast cycle. Note that the model is similar to [12][13][15] with a graceful extension for the multi-channel data broadcast model.

B. Abstraction of MCTA Traffic Awareness

Fig. 2 depicts the basic flowchart of MCTA which comprises two parts: the pull and the push procedures. The pull procedure responds to the pull requests for cold data, and the push procedure maintains a compound broadcast program. Since a client can submit an impatient pull request, the server has the exact number of regular pull requests for each cold item and the number of impatient requests for each hot item. The server deliberately makes *a single-item broadcast miss* by deferring the broadcast of a selected item for a cycle time. Meanwhile, the pull procedure can count the exact number of requests for this deferred item as *a reflective base*. Then, the reflection estimation can calculate the access frequencies of hot items according to the reflective base. Therefore, the server is aware of dynamic access frequencies of all broadcast items.

C. Pull Access Calculation

During a cycle time, the server can calculate exactly the mean access frequency $\overline{\lambda_i}$ of each cold item d_i in U^c as

$$\overline{\lambda_i} = \sum\nolimits_{1 \leq j \leq L} \frac{\lambda_i(j)}{L} = \frac{\lambda_i}{L}, \quad k+1 \leq i \leq m. \tag{1}$$

Hence, the server can obtain the pull access frequency distribution, denoted as $\Phi_{pull}(x) =< \overline{\lambda_{k+1}}, \overline{\lambda_{k+2}}, ..., \overline{\lambda_m} >$.

D. Push Access Estimation

The procedure has four phases: (1) client impatience modeling, (2) the selective deferment and reflective estimation, (3) estimation enhancement with the trigonometric tuning, and (4) the mergence of estimations on multiple push channels.

1) Client Impatience Modeling: Fig. 3 depicts two cases regarding a client's push access on P_x. Suppose a client with interest in d_i^x arrives at the j-th slot. In the case (1), this client has to wait $(i-1) - j + \frac{1}{2}$ slots where $\frac{1}{2}$ is an average access delay in the j-th slot. In the case (2), the wait time is $(L + i - 1) - j + \frac{1}{2}$ slots since the client has to wait until its broadcast in the next cycle. Let a random variable x present a client's patience in an exponential distribution with a mean patience ω. The probability density function of x is $f_\gamma(x) = \frac{1}{\omega} e^{-\frac{1}{\omega}x}$ where $0 \leq x < \infty$. Accordingly, the probability that a client will generate an impatient pull request for d_i^x is

$$\mathcal{F}_\gamma(i,j) = \left\{ \begin{array}{ll} 1 - e^{-\frac{1}{\omega}(i-j-\frac{1}{2})}, & i > j, \\ 1 - e^{-\frac{1}{\omega}(L+i-j-\frac{1}{2})}, & i \leq j. \end{array} \right. \tag{2}$$

Let $\lambda_i^x(j)$ be the access rate for d_i^x in the j-th slot. The server has the aggregate of impatient requests in a cycle time as $\gamma_i^x = \sum_{1 \leq j \leq L} \lambda_i^x(j) \cdot \mathcal{F}_\gamma(i,j)$. Observe that if every client

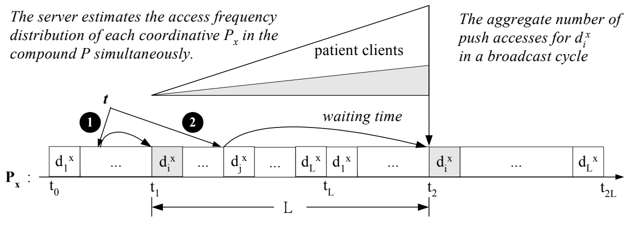

The server estimates the access frequency distribution of each coordinative P_x in the compound P simultaneously.
patient clients
The aggregate number of push accesses for d_i^x in a broadcast cycle
t
1
2
waiting time
P_x :
d_1^x
...
d_i^x
...
d_j^x
...
d_L^x
d_1^x
...
d_i^x
...
d_L^x
t_0
t_1
t_L
t_2
t_{2L}
L

Fig. 3. The push estimation procedure in a coordinative P_x.

TABLE I
SIMULATION PARAMETERS DESCRIPTION

symbol	meaning	value
m	the number of data items	1000, 10000
c	the number of channels	10, 50, 100
b	channel bandwidth	100, 500, 1000 KB
λ	access rate	10 ~100
θ	Zipf's skew coefficient	0 ~2.0
μ	channel service rate (b/s)	variable
ω	mean client patience	500, 1000, 10000

has an equal mean patience ω, $\sum_{1\leq j\leq L}\mathcal{F}_\gamma(i,j)$ is the same for each push item. We then have

$$\gamma_i^x = \sum\nolimits_{1\leq j\leq L}\overline{\lambda_i^x}\cdot\mathcal{F}_\gamma(i,j) = \overline{\lambda_i^x}\cdot(L - e^{-\frac{L^2}{2\omega}}). \quad (3)$$

Since $(L - e^{-\frac{L^2}{2\omega}})$ is constant, we have the ratio equation below.

$$\gamma_1^x : ... : \gamma_i^x : ... : \gamma_k^x = \overline{\lambda_1^x} : ... : \overline{\lambda_i^x} : ... : \overline{\lambda_k^x}. \quad (4)$$

2) Selective Deferment and Reflective Estimation: This phase estimates an access frequency distribution from (4). As depicted in Fig. 3, the server selects a single item d_i^x in each P_x and intentionally defers its broadcast for a cycle time. The selective deferment can compel the clients to disclose their interests in d_i^x. For those clients who are interested in d_i^x and arrive after t_1, they will be satisfied at t_2, except some clients whose patience is less than the waiting time $t_2 - t$. In $[t_1, t_2]$, the server will receive $\gamma_i^x = \sum_{1\leq j\leq L}\lambda_i^x(j)\cdot\mathcal{F}_\gamma(i,j)$ impatient requests for d_i^x. In addition, those patient clients will submit regular pull requests for d_i^x when they find that d_i^x is absent at t_2. Therefore, the server obtains the exact access frequency λ_i^x by adding γ_i^x and the number of the regular pull requests for d_i^x. Likewise, other impatient frequencies $< \gamma_1^x, \gamma_2^x, ..., \gamma_k^x >$ are available. With a reflective base λ_i^x in $[t_1, t_2]$, it is able to estimate the access frequency of each push item from (4) as

$$\overline{\lambda_j^x} = \frac{\gamma_j^x}{\gamma_i^x}\cdot\frac{\lambda_i^x}{L} = \frac{\gamma_j^x}{\gamma_i^x}\cdot\overline{\lambda_i^x}, \quad 1\leq j\leq k. \quad (5)$$

Thus, the server can obtain the dynamic access frequency distribution of each P_x, denoted as $\Phi_{push}^x(x) = <\frac{\gamma_1^x}{\gamma_i^x}\cdot\overline{\lambda_i^x}, ..., \frac{\gamma_{i-1}^x}{\gamma_i^x}\cdot\overline{\lambda_i^x}, \overline{\lambda_i^x}, \frac{\gamma_{i+1}^x}{\gamma_i^x}\cdot\overline{\lambda_i^x}, ..., \frac{\gamma_k^x}{\gamma_i^x}\cdot\overline{\lambda_i^x}>$.

3) Estimation Enhancement with Trigonometric Tuning: MCTA has the theoretical and quantitative analyses inherent in the SDR technique [5] as follows.

Theorem 1 *$\Phi_{push}^x(x)$ with a reflective base d_i^x has at least a cross with $\Psi_{push}^x(x)$ at $\Phi_{push}^x(i) = \Psi_{push}^x(i)$.*

Theorem 2 *With a reflective base d_i^x whose λ_i^x/γ_i^x is the smallest, we have $\lambda_i^x = \Phi_{push}^x(i) \lessapprox \Psi_{push}^x(i)$ if λ_i is continuously increasing, or $\lambda_i^x = \Phi_{push}^x(i) \gtrapprox \Psi_{push}^x(i)$ if λ_i is continuously decreasing.*

Property 1 *Given a flat P_x scheduled in order of access frequency, the item of the smallest λ_i^x/γ_i^x is either $d_{\lceil L/2\rceil}$ or close to $d_{\lceil L/2\rceil}$. Otherwise, the server can sort these items by their γ_i^xs and take $d_{\lceil L/2\rceil}$ as the reflective base.*

Accordingly, we further devise a tuning method based on the semi-monotonic feature of trigonometric functions to enhance the estimation. Given a P_x of L item slots, each slot is mapped to an angle range of $[0, \pi]$ in radians subsequently. For instance, the angle of d_1^x is 0, the angle of $d_{\lceil L/2\rceil}^x$ is $\frac{\pi}{2}$, and other d_i^x has its angle δ_i as $i\cdot\frac{\pi}{L}$ where $1\leq i\leq L$. Then, we have a tuned distribution $\Phi_\triangle^x(x)$ as

$$\Phi_\triangle^x(x) = \cos^2\delta_x\cdot\Phi_1^x(x) + \sin^2\delta_x\cdot\Phi_{\lceil L/2\rceil}^x(x), \quad (6)$$

where $\cos^2\delta_x + \sin^2\delta_x = 1$. Because $\Phi_{\lceil L/2\rceil}^x(x)$ crosses $\Psi_{push}^x(x)$ at $d_{\lceil L/2\rceil}$, we apply sine function to $\Phi_{\lceil L/2\rceil}^x(x)$. Likewise, we apply cosine function to $\Phi_1^x(x)$ which crosses $\Psi_{push}^x(x)$ at d_1. Hence, we can tune the estimation in the range between $\Phi_1^x(x)$ and $\Phi_{\lceil L/2\rceil}^x(x)$ with two cross points at the point $\delta_x = \frac{\pi}{2}$ where $\Phi_\triangle^x(\frac{L}{2}) = \Phi_{\lceil L/2\rceil}^x(\frac{L}{2}) = \Psi_{push}^x(\frac{L}{2})$ and at the point $\delta_x = 0$ with $\Phi_\triangle^x(1) = \Phi_1^x(1)$.

4) Mergence of Multiple-Channel Estimations: In a divide-and-consquer strategy, the task of estimating λ_i^xs of all push items is recursively divided into c_{push} smaller tasks which are solved simultaneously. The conquering phase integrates the answers. As $\Phi_{push}^1(x), ..., \Phi_{push}^{c_{push}}(x)$ are obtained, the push mergence performs a recursive merge-sort and achieves an overall $\Phi_{push}(x)$. Moreover, with two sorted lists, $\Phi_{pull}(x)$ and $\Phi_{push}(x)$, a classical merge-sort algorithm is used to generate a sorted, overall access frequency distribution $\Phi(x)$.

III. EXPERIMENTS AND RESULTS

A. Simulation Environment

Given a workload $\lambda = \sum_{1\leq i\leq m}\lambda_i$, each item has a distinct access frequency $\lambda_i = \lambda\cdot q_i$ in light of the Zipf distribution $q_i = (\frac{1}{i})^\theta / \sum_{1\leq i\leq m}(\frac{1}{i})^\theta$ [18]. The distribution is skewed gradually as the skew coefficient θ increases, but becomes uniform as $\theta = 0$. To simplify demonstration, we take the same reflective base in multiple P_xs. Three candidates for the reflective base: the first item (d_{FI}), the middle item (d_{MI}), and the last item (d_{LI}) in P_x, are inspected. Thus, $\Phi_{FI}(x)$, $\Phi_{MI}(x)$, $\Phi_{LI}(x)$ and $\Phi_\triangle(x)$ are compared with $\Psi(x)$ in terms of accuracy difference between the estimated and the accurate access frequencies of an item d_i where $d_i \in D$. Table I lists the simulation parameter description.

B. MCTA Performance Evaluation

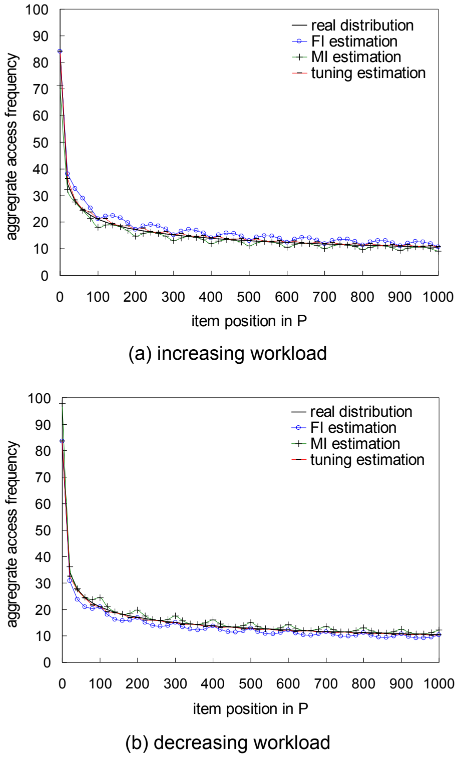
This subsection first inspects $\Phi_{FI}(x)$, $\Phi_{MI}(x)$, $\Phi_{LI}(x)$ and $\Phi_\triangle(x)$ under dynamic workload and access pattern. In addition, we examine the respective sensitivities of MCTA to dynamic workload, access pattern, and channel number.

1) Estimation under Dynamic Workload: As depicted in Fig. 4, $\Phi^x_{FI}(x)$ and $\Phi^x_{MI}(x)$ are "wavelike." The concatenation of $\Phi^x(x)$s results in $\Phi(x)$. Observe that $\Psi(x)$ is bounded by $\Phi_{FI}(x)$ and $\Phi_{MI}(x)$. In Fig. 4(a), when the workload increases, $\Phi^x_{FI}(x)$ is slightly above $\Psi(x)$, but $\Phi^x_{MI}(x)$ is below $\Psi(x)$. However, the positions of $\Phi^x_{FI}(x)$ and $\Phi^x_{MI}(x)$ are reverse under the decreasing workload. $\Phi^x_{FI}(x)$ is convex with a peak at d_{MI} as the workload increases; comparatively, $\Phi^x_{FI}(x)$ is concave, and d_{MI} becomes a foot as workload decreases. In contrast, $\Phi^x_{MI}(x)$ is a mirror rendering corresponding to $\Phi^x_{FI}(x)$ below $\Psi(x)$. Finally, it is promiment that $\Phi^x_{\triangle}(x)$ is very close to $\Psi(x)$.

The above phenomena are explained as follows. Depending on the reflective base in (5), the larger the value of γ^x_j/γ^x_i, the larger the value of λ^x_j is. Because d^x_{LI} is just the item ahead of d^x_{FI} cyclically, d^x_{FI} and d^x_{LI} face the same situation. The situation is symmetric with d^x_{MI} as the symmetric point. If an item is scheduled earlier within the first half of P_x, the rate of its push access to impatient requests can be more than those of other items scheduled lately. Hence, the estimated λ^x_j of d^x_j can be higher than the real value when d^x_{FI}, the item of the largest λ^x_i/γ^x_i, serves as the reflective base. Likewise, using d^x_{MI} whose λ^x_i/γ^x_i is the smallest as the reflective base can cause $\Phi^x_{MI}(x)$ to be lower than $\Psi(x)$. In the case of a decreasing workload, $\Phi^x_{FI}(x)$ and $\Phi^x_{MI}(x)$ in Fig. 4(b) are the inverted reflections of the corresponding curves in Fig. 4(a). The reason is that the rate of λ^L_{FI} to γ^L_{FI} is the smallest in a cycle, and thus the estimated access frequency for another item is lower than its real value.

2) Estimation under Dynamic Skew Access Pattern: There are several observation below. First, as depicted in Fig. 5(a), a hybrid data delivery model is profitable in the case of a higher skew pattern. Using parts of channels to broadcast fewer items of relatively high access frequency can reduce the mean access time. Oppositely, as shown in Fig. 5(b), a pure push delivery is beneficial under a less skew access pattern. Second, MCTA has high accuracy except when an item of a drastic change of access frequency is selected as the reflective base. In the context where item access frequencies change drastically, it is suggested that d_{MI} is a more suitable to avoid the performance degradation. Third, the effect of the trigonometric method is prominent. $\Phi_{\triangle}(x)$ is relatively closer to $\Psi(x)$ than $\Phi_{FI}(x)$ and $\Phi_{MI}(x)$ are. Consequently, MCTA with the trigonometric tuning method is able to attain very high accuracy.

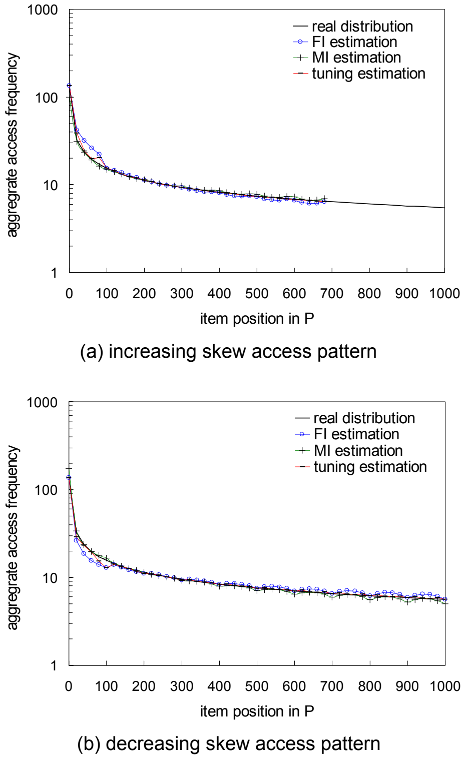
3) Sensitivity to Channel Number Variation: As shown in Fig. 6(a), the curves of $\Phi_{FI}(x)$, $\Phi_{MI}(x)$ and $\Phi_{\triangle}(x)$ go up slightly corresponding to the increase of channel number in the x-axis. However, we find that the standard deviations of accuracy difference go down. For example, the difference percents of $\Phi_{\triangle}(x)$ are 4.57%, 4.98%, 5.14%, 5.22%, 5.28%, 5.32%, but their respective deviations are 3.06, 0.70, 0.38, 0.18, 0.12, 0.09. It is because the gradient of item classification is coarse in case of fewer channels and can be refined with more channels. MCTA can attain a higher accuracy in a small P_x because the access frequencies of these items are close. Though the fast increase of workload can cause some items having drastic changes of access frequencies, the low accuracy difference as depicted in Fig. 6(a) presents the robustness of



(a) increasing workload

(b) decreasing workload

Fig. 4. MCTA under dynamic workload (baseline: $m = 1000$, $w = 500$, $c = 10$, $\theta = 0.3$, $\lambda = 100 \leftrightarrow 200$).



(a) increasing skew access pattern

(b) decreasing skew access pattern

Fig. 5. MCTA under dynamic access pattern (baseline: $m = 1000$, $w = 500$, $c = 10$, $\lambda = 100$, $\theta = 0.3 \leftrightarrow 0.6$).

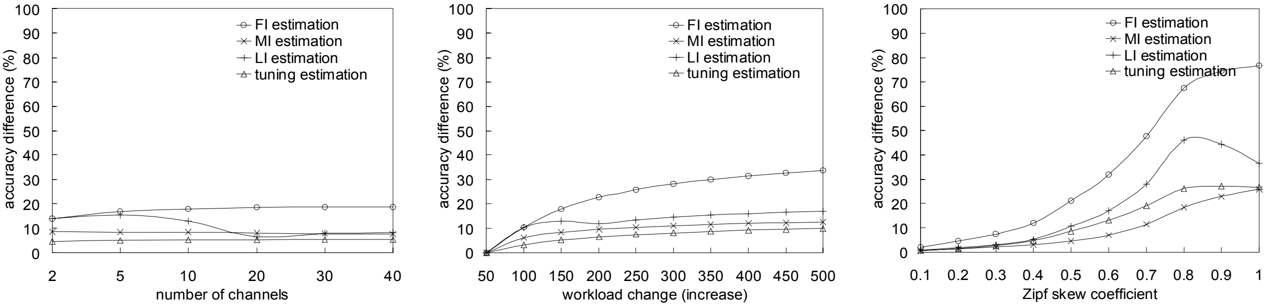


Fig. 6. Sensitivity of MCTA to traffic factors: (a) $m = 1000$, $c \cdot \mu = 100$, $\theta = 0.5$, $\lambda = 50 \leftrightarrow 150$, (b) $m = 1000$, $c \cdot \mu = 100$, $\theta = 0.5$, $\lambda = 50 \leftrightarrow 500$, and (c) $m = 1000$, $c \cdot \mu = 100$, $\lambda = 150$, $\theta = 0 \leftrightarrow 1.0$.

MCTA to the variance of channel number.

4) Sensitivity to Workload Variation: Fig. 6(b) depicts the scalable performance of MCTA. The accuracy difference increases slowly corresponding to the incremental workload change labeled in the x-axis. We observe that an increasing difference ascribes to the drastic change of workload in few P_xs which contain relatively hotter items. Selecting a reflective base with a drastic change of access frequency will increase the accuracy difference. It is noteworthy that the accuracy reduction is confined to a few P_xs. Hence, MCTA is still robust with a reliable reflective base, i.e., d_{MI}. Particularly, under a drastic workload change as $\lambda = 50 \rightarrow 500$, $\Phi_{\Delta}(x)$ is about 91%, and $\Phi_{MI}(x)$ is about 88%. This experimental results in essence conform to the general agreement that data broadcast is beneficial to the scalability problem.

5) Sensitivity to Access Pattern Variation: As depicted in Fig. 6(c), the less drastic the access pattern changes, the higher accuracy MCTA has. It is interesting to find that $\Phi_{MI}(x)$ is closer to $\Psi(x)$ than others when θ varies from 0 to a higher value. Explicitly, the items near the head and the rare of each P_x such as d_{FI} or d_{LI} can have higher variations in access frequency. Thus, $\Phi_{FI}(x)$ affects poorly the performance of the trigonometric tune. Comparatively, d_{MI} is less susceptible in this context where the access pattern has a drastic change. In addition, the curves go up smoothly or downward, especially while the access pattern becomes very skewed (θ from 0 to ≥ 0.8). As measured, most workload focuses on a few items of very high access frequencies. The estimated frequencies of these items account for most of accuracy difference. Oppositely, other items have lower estimation difference.

IV. Conclusions

The recent multi-channel data dissemination model has attracted many researchers' studies. In this paper, we have proposed an on-line multi-channel traffic awareness (MCTA) in response to the dynamic traffic changes in the multi-channel data dissemination environment. MCTA is able to periodically estimate the dynamic access frequencies of all items disseminated in the push channels. Extensive simulations have shown that the estimated access frequency distribution has a low inaccuracy as compared to the real one, consequently presenting the feasibility and robustness of MCTA.

Acknowledgement

The authors are supported in part by the Ministry of Education Project No. 89-E -FA06-2-4, and the National Science Council Project No. NSC 92-2213-E-002-001 and NSC 92-2213-E-002-010, Taiwan, Republic of China.

References

[1] S. Acharya, et al. Broadcast disks: Data management for asymmetric communications environments. In *Proceedings of ACM SIGMOD'95*, pages 199–210, May 1995.

[2] S. Acharya, et al. Balancing push and pull for data broadcast. In *Proceedings of ACM SIGMOD'97*, May 1997.

[3] M. H. Ammar and J. W. Wong. On the optimality of cyclic transmission in teletext systems. *IEEE Transactions on Comunications*, COM-35(11):1159–1170, 1987.

[4] C.-L. Hu and M.-S. Chen. Adaptive balanced hybrid data delivery for multi-channel data broadcast. In *Proceedings of IEEE ICC'02*, April 2002.

[5] C.-L. Hu and M.-S. Chen. Dynamic data broadcasting with traffic awareness. In *Proceedings of IEEE ICDCS'02*, 2002.

[6] Q. Hu, D. L. Lee, and W.-C. Lee. Dynamic data delivery in wireless communication environments. In *Proceedings of ER'98 Workshops on Mobile Data Access*, volume 1552 of *LNCS*. Springer, 1998.

[7] J.-L. Huang, W.-C. Peng, and M.-S. Chen. Binary interpolation search for solution mapping on broadcast and on-demand channels in a mobile computing environment. In *Proceedings of ACM CIKM'01*, 2001.

[8] T. Imielinski, S. Viswanathan, and B. R. Badrinath. Energy efficient indexing on air. In *Proceedings of ACM SIGMOD'94*, 1994.

[9] T. Imielinski, S. Viswanathan, and B. R. Badrinath. Data on air: Organization and access. *IEEE TKDE*, 9(3):353–372, May/June 1997.

[10] S. Jiang and N. H. Vaidya. Scheduling data broadcast to "impatient" users. In *Proceedings of ACM MobiDE'99*, pages 52–59, August 1999.

[11] R. Kravets and P. Krishnan. Power management techniques for mobile communication. In *Proceedings of ACM MOBICOM'98*, October 1998.

[12] W.-C. Lee, Q. Hu, and D. L. Lee. A study on channel allocation for data dissemination in mobile computing environments. *ACM/Baltzer MONET*, 4(2):117–129, 1999.

[13] W.-C. Peng and M.-S. Chen. Dynamic generation of data broadcasting programs for a broadcast disk array in a mobile computing environment. In *Proceedings of ACM CIKM-2000*, November 2000.

[14] K. Prabhakara, K. A. Hua, and J. Oh. Multi-level multi-channel air cache designs for broadcasting in a mobile environment. In *Proceedings of IEEE ICDE-2000*, pages 167–176, February 2000.

[15] K. Stathatos, et al. Adaptive data broadcast in hybrid networks. In *Proceedings of VLDB'97*, pages 326–335, 1997.

[16] C.-J. Su, et al. Broadcast scheduling for information distribution. *ACM/Baltzer Wireless Networks*, 5(2):137–147, 1999.

[17] J. X. Yu, T. Sakata, and K.-L. Tan. Statistical estimation of access frequencies in data broadcasting environments. *ACM/Baltzer Wireless Networks*, 22(6):89–98, 2000.

[18] G. K. Zipf. *Human Behaviour and the Principle of Least Effort*. Addison-Wesley, Reading, MA, 1949.