

## 테일러합렬에 의한 예측정보를 리용한 캐쉬교환 알고리즘에 대한 연구

한경일, 조성순

위대한 령도자 김정일동지께서는 다음과 같이 지적하시였다.

《우리는 우리 식의 프로그램을 개발하는 방향으로 나가야 합니다.》(《김정일선집》 제21권 증보판 42페이지)

현재 세계적으로 인터넷이 매우 빠른 속도로 증가되고 사회적으로 망봉사에 대한 수요가 급격히 높아짐에 따라 사용자요청에 대한 응답시간을 줄이고 대역폭을 절약하는 문제가 중요한 요구로 나서고있다.

웹브캐쉬화는 이러한 문제를 해결하는데서 관건적인 역할을 하는 기술로서 세계의 많은 과학연구기관들에서 다양한 캐쉬화기술들을 개발하고있으며 그 효과성은 실지 망환경에서 이미 검증되였다.

웹브캐쉬화는 봉사가 의뢰기요청에 대한 봉사내용을 캐쉬에 보관하였다가 의뢰기가 다시 그 내용을 요청하는 경우 캐쉬에서 직접 봉사하도록 함으로써 사용자대기시간을 줄이고 대역폭을 감소시키는 기술이다.[3]

일반적으로 웹브캐쉬화는 대리봉사기들에서 많이 리용되고있으며 여기서 가장 핵심적인 부분은 캐쉬교환알고리즘으로서 일명 캐쉬화알고리즘이라고도 한다.

전통적인 캐쉬교환알고리즘들은 의뢰기가 접근하였던 객체의 접근빈도수나 마지막 참조시간, 객체크기와 같은 정보에 기초하여 제거할 객체를 선택하는데 이것은 과거의 정보에 국한되므로 성능상 제한을 받고있다.[2]

캐쉬가 보관해야 할 객체들은 본질상 가까운 앞날에 사용자들에 의하여 접근될수 있는 객체들이므로 캐쉬교환에서 기본은 사용자요청을 정확히 예측하는것이다.

현재 캐쉬교환알고리즘들의 연구방향은 접근되였던 웹브객체들에 대한 예측정보를 리용하는데로 지향되고있으며 예측정보를 추출하는데 테일러합렬에 의한 근사값계산방법을 비롯한 여러 방법들이 리용되고있다.

논문에서는 객체의 과거접근정보와 테일러합렬에 의한 예측정보를 리용하는 보다 혁신적인 캐쉬교환알고리즘 TP-GDSF#(Taylor series Prediction-GreedyDualSizeFrequency#)를 제안한다.

### 1. 테일러합렬에 의한 예측정보추출

가장 많이 쓰이고있는 LRU, GD-Size, GDSF와 같은 캐쉬교환알고리즘들은 가장 최근에 접근된 페이지들의 정보를 가지고 제거할 객체를 선택하는데 리용된다. 이러한 정보들로서 접근빈도수, 마지막 참조시간, 화일크기, 객체수명 등을 들수 있다.

예측정보는 사용자에게 의하여 생성된 이러한 과거정보를 가지고 추출해야 한다.

문문에서는 고전력학에서 리용되는 《가속》(acceleration)의 개념을 리용하여 객체의 접근시간들로 예측정보를 추출하였다. 즉 객체의 과거의 접근시간간격이 점점 줄어든다면 그 객체는 점점 가속화되어 접근된다고 리해할수 있다.

다시말하여 시간차  $\Delta T = T_p - T_c$  ( $T_p$ : 예측시간,  $T_c$ : 현재시간)를 계산하여 이 값이 작은 객체는 보다 가까운 앞날에 접근될수 있으며 이 값이 큰 객체는 보다 후에 접근된다는 것이다.

문문에서는 이  $\Delta T$  인자를 세계적으로 가장 성능높은 캐쉬교환알고리즘으로 인정받고 있는 GDSF를 강화한 GDSF#알고리즘에 추가하여 새로운 캐쉬교환알고리즘을 제안한다.

GDSF#에서는 사용자에게 의하여 요청된 객체가 캐쉬에 보관될 때 식 (1)과 같이 열쇠값을 계산하여 객체우선권을 결정하며 그 순위에 따라 제일 작은 열쇠값을 가진 객체들이 제거대상으로 선택된다.[1]

$$K_i = L + F_i^\lambda \cdot C_i / S_i^\delta \quad (1)$$

여기서  $F_i$ 는 객체  $i$ 에 대한 접근회수,  $C_i$ 는 객체  $i$ 를 봉사기로부터 추출하는데 걸린 비용,  $S_i$ 는 객체  $i$ 의 크기,  $L$ 은 캐쉬수명이며  $\lambda, \delta$ 는 GDSF알고리즘을 강화하기 위하여 GDSF#알고리즘에서 추가한 상수로서  $\lambda=2, \delta=0.9$ 일 때 가장 높은 성능을 발휘한다.

새로운 캐쉬교환알고리즘에서는 다음과 같은 식을 리용하여 객체의 우선권을 결정한다.

$$K_i = L + \frac{F_i^\lambda \cdot C_i}{S_i^\delta \cdot \Delta T} \quad (2)$$

문제는 예측시간을 보다 정확하게 계산하는것이므로  $T_p$ 를 계산하는 방법을 보기로 한다.

매 페이지에 대하여 그것의 참조시간렬을  $T(1), T(2), \dots, T(n)$ 이라고 정의하면  $T(n)$ 은 간단히 객체의  $n$ 번째 참조시간이다.

이때 페이지의  $n+1$ 번째 참조시간  $T(n+1)$ 을 예측하는데 테일러합렬을 리용하는데  $T(n+1)$ 을 대략 예측하는데 리용되는 두마디테일러다항식에 대한 테일러합렬표현식은 다음과 같다.

$$T(n+1) \approx T(n) + \frac{T'(n)}{1!} + \frac{T''(n)}{2!} \quad (3)$$

식 (3)에서

$$T'(n) = \frac{T(n) - T(n-k)}{k}, \quad T''(n) = \frac{T'(n) - T'(n-k)}{k} = \frac{T(n) - 2T(n-k) + T(n-2k)}{k^2}$$

이다.

이제  $k=1$ 로 설정하고  $T(n)$ 을 현재시간  $T_c$ 라고 하자. 그리고  $T_{c-1}$ 을 객체가 가장 마지막에 접근되었던 시간,  $T_{c-2}$ 를 객체가  $T_{c-1}$ 시각전에 접근되었던 시간이라고 하자. 그러면 예측된 시간은 다음과 같이 된다.

$$T_p \approx T_c + \frac{(T_c - T_{c-1})}{1!} + \frac{(T_c - T_{c-1}) - (T_{c-1} - T_{c-2})}{2!} = (5T_c - 4T_{c-1} + T_{c-2})/2 \quad (4)$$

식 (4)로부터 객체가 사용자에게 의하여 접근될 때마다 열쇠값을 계산하여 우선권을 재할당한다.

## 2. 예측정보를 리용한 캐쉬 교환알고리즘

테일리합렬에 기초하여 객체의 다음번 접근시간에 대한 예측정보를 추출하여 캐쉬 교환에 리용하는 새로운 알고리즘(TP-GDSF#)은 그림 1과 같다.

론문에서 제안한 새로운 캐쉬교환알고리즘을 대리봉사기프로그램 Squid-3.1.1에 구현하여 이전의 캐쉬교환알고리즘(LRU, LFUDA, GDSF)들과 성능대비분석을 위한 실험을 진행하였다.

실험에서는 성능측정지표인 제안한 알고리즘명중률(Hit Rate)과 바이트명중률(Byte HitRate)을 비교하였다.(그림 2)

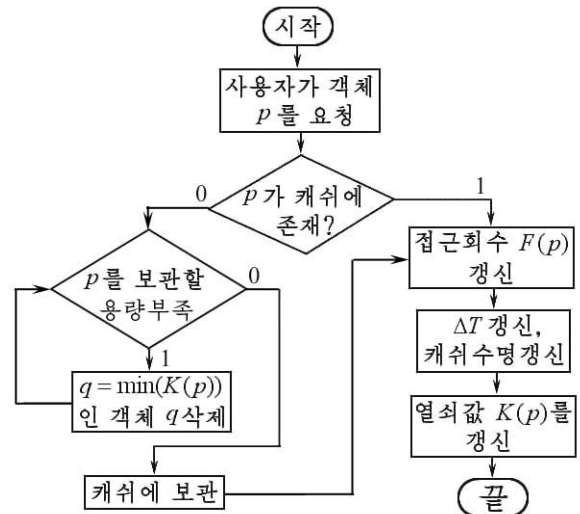


그림 1. 테일리합렬에 의한 예측정보를 리용한 캐쉬교환알고리즘

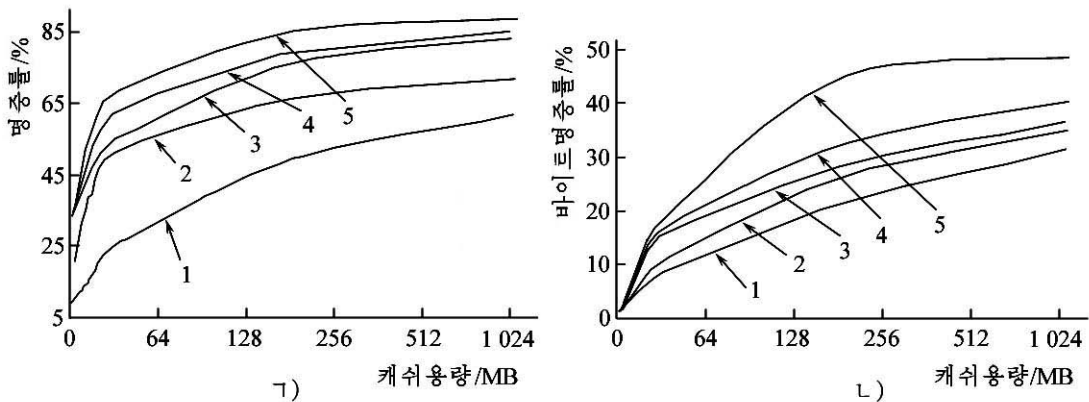


그림 2. 캐쉬교환알고리즘들의 명중률(H)과 바이트명중률(L)비교

1-LRU, 2-GD-Size, 3-LFUDA, 4-GDSF, 5-제안한 방법

그림 2로부터 론문에서 제안한 알고리즘의 명중률과 바이트명중률이 제일 높다는것을 알수 있다.

## 맺 는 말

과거의 접근정보들만을 가지고 캐쉬교환을 하던 종전의 알고리즘과 달리 앞으로 접근 될 시간을 예측하여 효과적으로 교환을 진행하는 알고리즘을 새롭게 제기하여 대리봉사기의 성능을 높임으로써 사용자대기시간을 줄이고 대역폭을 훨씬 절약하도록 하였다.

## 참 고 문 헌

- [1] J. B. Patil et al.; International Journal of Computer Science and Applications, 5, 4, 1, 2008.
- [2] K. H. Yeung et al.; Proceedings of the 22<sup>nd</sup> Conference on Local Computer Networks, 11, 184, 1997.
- [3] L. Breslau et al.; In Proceedings of Conference on Computer Communications(IEEE Infocom), 1, 126, 1999.

주체 103(2014)년 3월 5일 원고접수

## **On Cache Replacement Algorithm using the Prediction Information by Taylor Series**

*Jo Song Sun, Han Kyong Il*

We proposed a new cache replacement algorithm called Taylor Series Prediction Greedy Dual Size Frequency#.

This algorithm allows to keep the object having the suitable cache key value referring predictable time for the next access of the objects.

Key words: Taylor series, predictable information