ARTICLE IN PRESS

Future Generation Computer Systems ▮ (▮▮▮▮) ▮▮▮–▮▮▮

Contents lists available at ScienceDirect

Future Generation Computer Systems

journal homepage: www.elsevier.com/locate/fgcs

# Efficient cache resource aggregation using adaptive multi-level exclusive caching policies

Yuxia Cheng [a,*], Yang Xiang [a], Wenzhi Chen [b], Houcine Hassan [c], Abdulhameed Alelaiwi [d]

[a] *Deakin University, 221 Burwood Highway, Burwood, VIC, 3125, Australia*
[b] *Zhejiang University, Zheda Road 38, Xihu District, Hangzhou, China*
[c] *Polytechnic University of Valencia, Camino de Vera, s/n 46022, Valencia, Spain*
[d] *King Saud University, Riyadh 11543, Saudi Arabia*

## HIGHLIGHTS

- Propose a new multi-level exclusive cache policy.
- Design a new local Reuse Distance based Adaptive Replacement Caching (ReDARC) Algorithm.
- Design a new distributed Adaptive Level-Aware Caching Algorithm (ALACA).

## ARTICLE INFO

*Article history:*
Received 1 December 2016
Received in revised form 1 August 2017
Accepted 16 September 2017
Available online xxxx

*Keywords:*
Buffer cache
Multi-level
Exclusive caching
Adaptive policy

## ABSTRACT

Multi-level buffer cache hierarchies are now commonly seen in most client/server cluster configurations, especially in today's big data application deployment. However, multi-level caching policies deployed so far typically use independent cache replacement algorithms in each level, which has two major drawbacks: (1) File blocks may be redundantly cached on multiple levels, reducing the actual aggregate cache usable size; (2) Less accurate replacement decisions at lower level caches due to weakened locality. Inefficient cache resource usage may result in noticeable performance degradation for big data applications.

To address these problems, we propose new adaptive multi-level exclusive caching policies that can dynamically adjust replacement and placement decisions in response to changing access patterns. (1) First, to capture locality information in multi-level cache hierarchies, we propose a Reuse Distance based Adaptive Replacement Caching (ReDARC) algorithm that adopts reuse distance as the means of locality measure and adaptively balances between the Small Reuse Distance (SRD) set and Large Reuse Distance (LRD) set. (2) Second, to achieve exclusive caching and make global caching decisions, we propose an Adaptive Level-Aware Caching Algorithm (ALACA) that works collaboratively with ReDARC. The ALACA algorithm uses an adaptive probabilistic PUSH technique that allows lower caches to push blocks to higher caches and appropriately decide blocks' caching locations with the ReDARC algorithm. In this way, we achieve multi-level exclusive caching with significant cache performance improvement. Our trace-driven simulation experiments show that the policies we proposed achieve a reduction of the client average response time of 8 percent to 56 percent over other multi-level cache schemes.

© 2017 Elsevier B.V. All rights reserved.

## 1. Introduction

As the performance gap between processors and storage systems grows, memory cache plays an important role in the overall system performance. Buffer cache in computing systems is used to provide quick access to recently or frequently used data resides in storage devices [1]. In modern data centers, high-end storage systems typically have tens or even hundreds of gigabytes of cache RAM [2] to improve I/O performance, and clients of storage systems also have several gigabytes of memory for caching. All these buffer caches form an increasingly large amount of memory capacity.

Although storage caches and client caches become larger and larger, they do not achieve the performance commensurate to their aggregate cache resource [3]. Therefore, how to effectively and efficiently utilize this buffer cache resource becomes an important issue. Especially in today's service-oriented cloud computing environment, multi-level buffer cache hierarchies are commonly seen in most client/server cluster systems [4]. For example, in the data

---

[*] Corresponding author.
*E-mail address:* yuxia.cheng@deakin.edu.au (Y. Cheng).

http://dx.doi.org/10.1016/j.future.2017.09.044
0167-739X/© 2017 Elsevier B.V. All rights reserved.

Please cite this article in press as: Y. Cheng, et al., Efficient cache resource aggregation using adaptive multi-level exclusive caching policies, Future Generation Computer Systems (2017), http://dx.doi.org/10.1016/j.future.2017.09.044.

center scenario, a typical application server uses local DRAM as first level cache, and connected to a remote high-end SSD storage server as the second level cache, and finally connected to a remote large volume storage server as the whole data storage medium. We are facing the challenge of effectively managing large amounts of I/O buffer caches and multi-level cache hierarchies [5].

In a traditional three-tier Web server configuration, each server tier has large buffer caches which is typically seen in big data applications. Most multi-tier server configurations deployed so far use independent buffer cache management policies at each level. Previous research [6] pointed out that single-level cache replacement algorithms perform unsatisfactory when used in the multi-level cache hierarchies. Therefore, inefficient management of these caches may result in noticeable performance degradation for big data applications. In this paper, we propose the new universal hierarchy-aware multi-level cache management policies that collaboratively and efficiently manage buffer cache hierarchies to achieve better overall cache performance.

### 1.1. *Challenges in multi-level caching*

Previous researches have focused on efficient management of buffer caches to improve system performance. The single-level cache replacement algorithms were extensively studied for decades. The LRU algorithm is widely used due to its simplicity and generality, and many new replacement algorithms, such as LRU-K [7], 2Q [8], LRFU [9], LIRS [10], ARC [11] etc., are proposed to address the deficiency of LRU in handling some sequential or looping data access patterns. However, single-level cache management policies deployed in multi-level cache hierarchies are considered inefficient. Multi-level caching faces the following two major challenges:

(1) The first challenge is the weakened locality in lower level caches. Caching is based on the temporal locality principle of data accessing, and only the first level cache can see the original access pattern. Lower level caches can only see the misses in higher level caches. Therefore, application's access requests, as seen by lower level caches, are filtered by the higher level caches. The result is that traditional single-level cache algorithms that manage each cache level independently cannot effectively manage original locality in multi-level cache hierarchies.

(2) The second challenge is data redundancy. In level independent caching, data may be cached redundantly by multiple cache layers. This results in smaller actually usable cache resources than the aggregated cache resources in multi-level cache hierarchies. Particularly when each level has comparable cache size, a large proportion of data blocks stored in lower level caches will not be accessed. For example, in a multi-level cache with the first-level of DRAM (128 GB), the second level of high-end pci-e SSD (500 GB), the third level of data SSD (1 TB).

### 1.2. *Our approach*

In this paper, we propose the new adaptive multi-level caching policies that dynamically adjust replacement and placement decisions in response to changing data access patterns.

First, we propose a single-level Reuse Distance based Adaptive Replacement Cache (ReDARC) algorithm. ReDARC adopts reuse distance as a means of locality measure rather than recency that is typically used in LRU-like cache algorithms, and divides accessed blocks into Small Reuse Distance (SRD) set and Large Reuse Distance (LRD) set. ReDARC adaptively balances the sizes of SRD and LRD sets according to evolving access patterns and guarantees that blocks in SRD set are all cached in memory. We use ReDARC to address the weakened locality challenge in multi-level cache hierarchies.

Second, we propose an Adaptive Level-Aware Caching Algorithm (ALACA) that works collaboratively with ReDARC. The ALACA algorithm uses an adaptive probabilistic PUSH technique that allows lower caches to push blocks to higher caches and appropriately decide blocks' caching locations with the ReDARC algorithm. Using a simple hint information provided by the ReDARC algorithm, the ALACA algorithm can adaptively make global caching decisions and achieve the multi-level exclusive cache property across the hierarchy. Through the efficient cooperation of ReDARC and ALACA algorithms (shortened as REAL policy), our solution (the REAL policy) can effectively address the data redundancy and weakened locality problems with low network traffic, disk I/O and computational overheads. The trace-driven simulation experiments show that the cache replacement policies we proposed achieve the improvement of the client average response time of 8 percent to 56 percent over other state-of-the-art multi-level cache schemes.

The rest of this paper is organized as follows: Section 2 discusses related work. Section 3 describes our design motivation of adaptive multi-level exclusive caching policies. Section 4 presents a detailed description of our proposed adaptive algorithms. In Section 5, we report trace-driven simulation results for performance evaluation and comparisons of several multi-level caching schemes. Finally, we conclude and discuss future work in Section 6.

## 2. Related work

Previous researchers have studied the single level cache replacement algorithms for decades. Belady's MIN [12] is an optimal off-line algorithm [13]. The classic LRU algorithm is widely used. When cache is full, LRU replaces the least recently used blocks. However, pure LRU only captures recency information and does not capture frequency information. Many new replacement algorithms, such as LRU-K [7], 2Q [8], LRFU [9], LIRS [10], ARC [11] and SOPA [14] are designed considering both recency and frequency and capturing various access patterns. These algorithms were proposed in the single level cache hierarchy and did not take multi-level cache hierarchies into account. Specifically, LIRS [10] uses the reuse-distance based algorithm while its cost of keeping ghost caches are high. ARC [11] leverages the adaptive technique to adjust frequently accessed blocks with two dynamic list while it considers little reuse distance metric. The ReDARC algorithm proposed in this paper combines the benefits of using reuse distance and adaptive adjustment.

Multi-level cache management faces more challenges than single level cache management. Multi-level caching policies need to address challenges like weakened locality in lower level caches and data redundancy among different cache levels. Zhou et al. [15] proposed MQ replacement algorithm that leverages multiple lists to keep frequently accessed blocks with long reuse intervals. MQ was designed to improve the second-level cache performance. Bairavasundaram et al. [16] presented the X-RAY mechanism that monitors metadata updates in the RAID controller cache, and uses the gray-box technique to obtain the information on the content of the upper level cache.

Wong and Wilkes [6] proposed a way to eliminate redundancy by applying a unified LRU algorithm to implement exclusive caching. They introduced a DEMOTE operation to transfer the data that was popped from the client cache to the storage cache. DEMOTE tries to improve the overall hit ratio over inclusive policy. However, the DEMOTE operation results in high network traffic and system overhead to DEMOTE the evicted pages. Eviction-based cache placement algorithm [17] is designed to reduce the network resource usage. Eviction-based policy uses a client content tracking (CCT) table to keep a client's (high level cache's) eviction information, and periodically sends the information to the lower

Please cite this article in press as: Y. Cheng, et al., Efficient cache resource aggregation using adaptive multi-level exclusive caching policies, Future Generation Computer Systems (2017), http://dx.doi.org/10.1016/j.future.2017.09.044.

Download English Version:

https://daneshyari.com/en/article/6873058

Download Persian Version:

https://daneshyari.com/article/6873058

Daneshyari.com