Sudan University of Science and Technology

College of Graduate Studies

إستخدام إنموذج العميل لتحسين تفريع ذاكرة الوكيل المخبأه

PROXY CACHE CLEANUP IMPROVEMENT BY USING AN AGENT-BASED MODEL (PCCIA)

A thesis Submitted For Awarding the PHD Degree in Computer Science

By: Hiba Ali Nasir Sirour

Supervisor: Dr. Yahia Abdalla Mohammed

Co Supervisor: Dr. Amir Abdelfattah Ahmed Eisa

September 2014

المستخلص

تعد ذاكرة الوكيل المخبأه واحدة من الحلول الفعالة لتجنب االاختناق فى خدمة الانترنيت و تقليل الإزدحام وتحسين كفاءة خدمة الإنترنيت. وتعتبر سياسات الاحلال هو أساس نظام الذاكرة. وصفت هذه الدراسة إستخدام إنموذج العميل الذكى لتحسين اداء الذاكرة المخبأه.تم تطوير نظام العملاء المتعددين للتحكم فى عملية تفريغ الذاكرة المخبأه للمعمارية الهرمية.

أستخدم المنطق الضبابى لدمج سياسات الاحلال Least Frequency Used, Least Recently Used and Size فى ذاكرة الوكيل الاب, وتم إستخدام سياسات Least Frequency Used and Least Recently Used فى جانب الذواكر الأبناء. بمبادره من العميل المفرغ للذاكرة تم إستخدام المنطق الضبابى لإتخاذ قرار ذكى لمسح صفحة الانترنيت التى لها اولوية تفريغ عالية. تم التنسيق بين العملاء المفرغي لذاكرة الأب والأبناء وذلك لانجاز مهمة التفريغ بطريقة كفؤة. وكل العملاء لهم هدف مشترك عام وهو زيادة نسبة عدد الطلبات ونسبة حجم الطلبات بالبايت.

قام الوكيل المنسق بتطبيق قواعد للتنسيق عندما تكون صفحة الإنترنيت الموجودة فى ذاكرة الأب والأبناء لها اولوية تفريغ متوسطة. تم تطبيق خوارزمية التعلم Q بواسطة العميل المفرغ لتجنب الحسابات عندما يصل العميل الى حالة مشابهه وذلك لإتخاذ الفعل المناسب.

تم إسناد قيمة لكل فعل. يصل العميل المفرغ الى أفضل فعل الذى يقود الى الحالة الهدفية التى لها قيمة عالية.وبقية الأفعال لها قيم منخفضة.تم تمثيل الحالات والافعال فى شكل عقد تمثل الحالات ,تم تمثيل الفعل بحركة العميل من عقدة لأخرى.

أستخدم خمس عينات مولدة بإستخدام المحاكى Webtraff لإختبار الإنموذج ,هذه العينات تمثل طلبات المستخدمين التى استخدمت فى حجم الذاكرة. أستخدمت مصفوف القياسات القياسية نسبة عدد الطلبات ونسبة حجم الطلبات بالبايت لتقييم أداء الذاكرة.

أوضحت نتائج المحاكى ان الطريقة الجديدة تنجز أفضل على سياسات الاحلال Least Frequency Used, Least Recently Used and Size فى نسبة عدد الطلبات ونسبة حجم الطلبات بالبايت.

ABSTRACT

Web proxy caching is one of the effective solutions to avoid web service bottleneck, reduce traffic over the Internet and improve scalability of the web service. The core of a caching system is the caching replacement policies. This study describes the use of intelligent agent model to improve the performance of the proxy cache. A multi-agent system has been developed to control the cache cleanup task on the hierarchy caches.

Fuzzy logic is used to combine LFU, LRU and Size caching replacement policies on the parent cache side. LFU and LRU policies are used on the child caches side. Cache cleaner Agents use fuzzy logic to make an intelligent decision and remove the web object proactively when it has high clean up priority. Reactive Coordination has been applied between the parent and child cleaner agents to achieve the cleanup task in efficient way, they have a common goal to increase hit ratio and byte hit ratio.

Coordination agent applied the coordination rules when the web object with medium priority is found in parent and children caches. Q-learning algorithm has been implemented by the cleaner agent to avoid difficult calculation when it reached a similar state and take a suitable action.

A reward value has been associated to each action, when Cleaner agent takes its optimal action that leads to the goal, it has an instant high reward. Other actions have low reward. States and actions had been represented on a graph each node represented a "state", agent's movement from one node to another represented the "action".

The model has been tested using five samples of workload generated using Webtraff simulator, these samples represented the users requests and used cache sizes. The standard performance metrics Hit Ratio and byte hit ratio are used to evaluate the cache performance.

Simulation results show that when the cache size increase the new approach PCCIA performs better than LRU,LFU and Size replacement polices in terms of hit rate and byte hit rate.