

発表概要

世代 GC 方式におけるキャッシュを意識した 殿堂入り時期調節手法の提案

吉川 隆 英[†] 田浦 健次朗^{††} 近山 隆^{†††}

世代 GC 方式において、新たに生成されたデータは、何回かの GC を経た後、旧世代領域に移される。これを「殿堂入り」という。殿堂入り時期が遅すぎると、長寿命データと新世代ゴミが長期間新世代メモリ領域に混在するため、ユーザプログラム中や新世代 GC 中でのキャッシュヒット率が低下する。逆に、殿堂入り時期が早すぎると、短寿命データが新世代領域で回収できず、新世代 GC の回収効率が悪化し、旧世代 GC が発生しやすくなる。殿堂入りの適切な時期は、プログラムによって、また 1 つのプログラムの中でもその実行フェーズによって異なる。これまで、プログラム実行時に動的に殿堂入り時期を調節する手法はいろいろ提案されてきたが、主に新世代 GC の回収効率を改善することに主眼がおかれており、キャッシュの効果が、動的な殿堂入り時期調節の基準に、実験データに裏打ちされる形で明快に反映されていなかった。そこで本研究では、まず、様々なプログラムにおいて、実際のデータ寿命分布、殿堂入り時期の違いによるキャッシュミス数と実行時間の測定を行い、メモリ領域中のデータの振舞いに対するモデルを作成した。そして、このモデルに基づく殿堂入り時期調節手法の提案を行った。また、この殿堂入り時期調節手法を、ヒープサイズを自動調節する世代 GC を行う並行並列論理型言語処理系 KLIC 上に実装し、動的に殿堂入り時期調節を行わない世代 GC との性能比較を行った。

A Cache-conscious Adaptive Tenuring Policy for Generational GC

TAKAHIDE YOSHIKAWA,[†] KENJIRO TAURA^{††}
and TAKASHI CHIKAYAMA^{†††}

In generational GC schemes, objects are allocated to the young generation area and are advanced to the old generation area after surviving a small number of collections. This advancement is called tenuring. Tenuring too late makes some short-lived objects that some of them have already become garbage and long-lived objects reside together in the young generation, making memory reference locality worse. On the other hand, tenuring too early makes it impossible to collect short-lived objects in the young generation; its mark/cons ratio becomes worse and, as short-lived objects are moved to the older generation, more older generation GCs will be required. For the best performance, we should adjust tenuring timings dynamically according to programs and their execution phases. Many adaptive tenuring policies have been proposed. However, most of them aim at improving mark/cons ratio of the younger generation and improvements in cache performance are not proven with experimental evidences. In this work, we (1) measure object lifetime distributions on several programs, and how cache misses and execution times vary with different tenuring timings, (2) make a simple analytical model to estimate an appropriate young generation size, (3) propose a cache-conscious adaptive tenuring policy, and (4) implement dynamic young generation size adjustment mechanism with this policy into KLIC and compare its execution time to one with conventional generational GC on several programs.

(平成 14 年 1 月 29 日発表)

[†] 東京大学大学院工学系研究科

Graduate School of Engineering, University of Tokyo

^{††} 東京大学大学院情報理工学系研究科

Graduate School of Information Science and Engineering, University of Tokyo

^{†††} 東京大学大学院新領域創成科学研究科

Graduate School of Frontier Sciences, University of Tokyo