発表概要

便宜的ガーベッジコレクションの並行化について

川又 晃† 羅 翔† 寺島元章†

当研究室で開発した便宜的ガーベッジコレクション（occasional GC）の並行化とその実装について述べる．便宜的 GC とは，リスト処理などの純計算側が消費したヒープ中の最新の使用域のみを対象として使用中オブジェクトの圧縮（sliding compaction）処理を行うものである．その効果として，GC 処理時間の短縮化と，ヒープ使用の節約に伴うワーキングセットの縮小で純計算時間の短縮も図れることが停止回収型の実装から示されている．この便宜的 GC の並行化は，毎回行われる GC 処理の実時間（realtime）化と定期的に行われる GC の並列（concurrent）化とからなる．前者は通常の掃除をこまめに行うことを，後者は時間のかかる大掃除を別個に行うことを意味する．GC の圧縮処理の実時間化に必要な，write-barrier と輪切り複写の機能は，そのまま並列化にも適用できる．並列化ではこれらに，排他制御が付加されるだけである．輪切り複写とは，一度に圧縮するオブジェクトを移動先にその複製が作られる範囲に限定することであり，GC の圧縮処理中での中断を可能にする．また，write-barrier はオブジェクト参照に伴う純計算側の負荷を減らす．純計算側は Lisp の一方言である PHL を用い，GC 側はスレッドライブラリを利用したコルーティンという汎用的な環境下で実装を行った．数種の Lisp プログラムの実行結果から得られた本 GC の実時間特性についても述べる．

Incremental Occasional Garbage Collection

Akira Kawamata,† Xiang Luo† and Motoaki Terashima†

The design and implementation of incremental occasional garbage collection are presented. The occasional garbage collection is a new type of garbage collection (GC) based on a "mark-and-compact" or sliding compaction scheme. The GC focuses its task of scavenging on most recently generated data objects to gain time, and its good performance is shown by a prototype of "stop-and-collect" version. The incremental occasional garbage collection is made up of two features: "real-time" and "concurrency". The former is performed usually, while the latter is periodically to gain more spaces. They need common schemes of a write barrier and a coherent copy. The write barrier is less costly in time for coordination than a read barrier, and the coherent copy makes each data object consistent upon its relocation. The concurrency needs exclusive control in addition to them, so that these two features are implemented effectively in the GC. The incremental GC is implemented on a multi-processor machine with shared memory by using (POSIX) thread library that makes it portable. The analysis of the GC on its performance is done by using our experimental data obtained from the execution of typical Lisp programs.

(平成 11 年 8 月 5 日発表)

† 電気通信大学情報システム学研究科
Graduate School of Information Systems, University of Electro-Communications