

発表概要

参照カウントに対応した ソフトウェアトランザクショナルメモリの実装

岡本 尚文^{1,a)} 南出 靖彦²

2013年3月1日発表

マルチスレッドプログラミングにおいて逐次処理と同様に並列処理を記述しつつ共有資源の保護を行う方法としてソフトウェアトランザクショナルメモリ (STM) が注目されている。本発表は C++ 言語による STM アルゴリズムの 1 つである Lazy Snapshot Algorithm (LSA) を用いたオブジェクトベースの STM ライブラリの実装について述べる。はじめに、抽象化された LSA は実際のオブジェクトベースで実装されるアルゴリズムとはかけ離れたものであるため、新たに object history という考えを導入して実装に対応したアルゴリズムの定式化を行う。LSA は Java 言語によるオブジェクトベースの実装と C 言語によるワードベースの実装がなされたものであり、本発表における実装は C++ 言語でオブジェクトベースの LSA を再現したため、C++ 言語にはないガーベジコレクションを参照カウント型のポインタで行っている。しかし、参照カウントによるガーベジコレクションでは循環参照のあるデータ構造に対してゴミの回収が行われないという問題がある。LSA で実装したライブラリ内部のデータ構造は並列処理によっては循環参照が発生する可能性を持つため、弱い参照を用いたライブラリ内部のデータ構造の変更による循環参照の解決方法を示す。

Implementation of Software Transactional Memory with Reference Counting

NAOFUMI OKAMOTO^{1,a)} YASUHIKO MINAMIDE²

Presented: March 1, 2013

Software transactional memory (STM), which enables to protect shared resources and write a concurrent program as a sequential, has recently been focus of interest in multithread programming. This presentation is about the implementation of an STM library in C++ using object-based lazy snapshot algorithm (LSA). Felber et al. presented LSA in an abstract form and implemented two versions based on it; one is object-based LSA in Java, and another is word-based LSA in C. We first introduce the concept of “object history” to refine the abstract algorithm to make it consistent with the object-based implementation. To translate object-based LSA into the implementation in C++, we use reference counting pointers instead of garbage collection in Java. However, the reference counting does not have the capability to collect memory in a circular reference, and data structures constructed inside the library may become circular structures. We show modification of the data structure with a weak reference to solve this problem.

¹ 筑波大学大学院システム情報工学研究科
Graduate School of Systems and Information Engineering,
University of Tsukuba, Tsukuba, Ibaraki 305-8573, Japan

² 筑波大学システム情報系
Faculty of Engineering, Information and Systems, University
of Tsukuba, Tsukuba, Ibaraki 305-8573, Japan

a) nokamoto@score.cs.tsukuba.ac.jp