

# UPPAALによる時間仕様検証

小山尾 登

タスクなどがいわゆる「三すくみ」により全く動かなくなることをデッドロックという。これに対し、一定の場所をループするようになることをライブロックという。ライブロックが起こらないことを検証するのは難しい。本稿ではモデル検査ツールUPPAALによってライブロックの検証を行う。  
(編集部)

## はじめに

本誌 2009 年 12 月号の並列プロセッサ XMOS の記事 (pp.141-148) で、並列プログラミング言語である Occam と CSP (Communicating Sequential Processes) の関係を説明する際に英国ケント大学教授の Peter Welch 氏が指摘した Java VM の問題について紹介した。その後、実際に同氏の Web ページからサンプル・プログラム<sup>(1)</sup>をダウンロードして実験してみたが、ライブロックは起きなかった。10 年以上経っているので、Java の最新版はライブネス問題に対応したのかと思った。しかし念のため、同氏にメールで確認すると、VM の実装仕様に依存するという。それで、ログをよく見ると待ち行列が FIFO ではなく LIFO になっているようだった。別の言い方をすると、待ち行列が待ちスタックになっているのだ (図 1)。ダウンロードした同氏のプログラムは FIFO でライブネス問題が発生するように作られているので、LIFO の VM では問題が発生しな

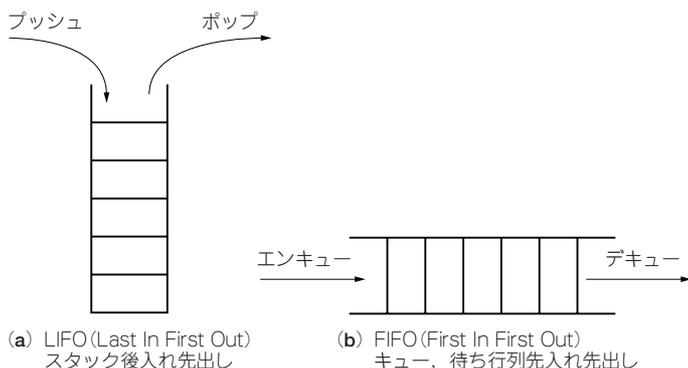


図1 LIFO/スタックとFIFO/キュー

注1: java version "1.6.0\_15", Java SE Runtime Environment (build 1.6.0\_15-b03), Java HotSpot Client VM (build 14.1-b02, mixed mode, sharing).

かったようだ。そこで、LIFO で問題が発生するようにサンプルを書き換えてみるとライブロックが見事に発生する。同氏が使っていた 10 年前の Java VM は FIFO なのに、最新版で<sup>注1</sup>は何と LIFO になっている。

LIFO ではリアルタイム性を保証できないし、リソース不足で簡単にライブロックが発生する。Java はガベージ・コレクションが始まると止まってしまうので組み込みでは使えない、と言われている。iPhone などではガベージ・コレクションを使っていないという話も聞く。しかし、待ち行列の扱いはそれ以前の基本的問題ではないだろうか。今回は、この問題を深く掘り下げたい。

## 1. ライブロックとライブネス

### ● ライブロックは出口のないループ

そもそもライブロックとは何か。デッドロックはシステムが停止してしまうことで、ライブロックは動いてはいるものの特定の動作のみで、ある動作が実行できなくなっている状態である。LTSA<sup>(2)</sup>のステート・マシンで表現する

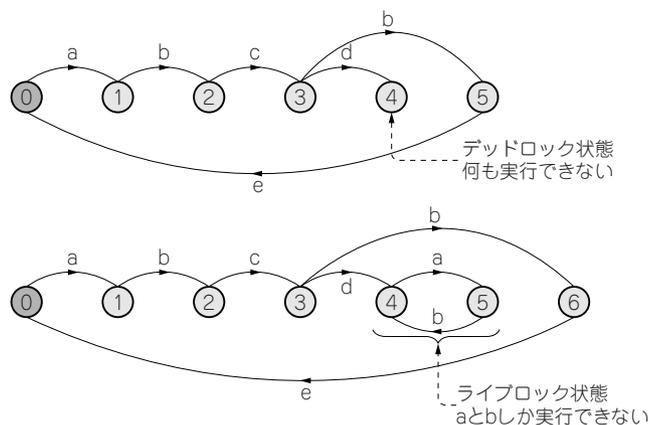


図2 デッドロックとライブロック