

## 並行処理プログラムのためのテスト支援ツールに関する一考察

1 J-4

片山徹郎\* 菰田敏行\* 古川善吾\*\* 牛島和夫\*

\*九州大学 工学部 情報工学科

\*\*九州大学 情報処理教育センター

## 1. はじめに

近年、LAN やマルチプロセッサの普及により、並行処理プログラムが実際の場で多く書かれるようになってきた。これに伴い、並行処理プログラムの信頼性向上の方法の一つとして、テストが重要な役割を演じてきている。逐次処理プログラムのテスト法については、様々な方法が提案され、テスト支援ツールも数多く開発されている。しかしながら、並行処理プログラムのテスト法については、十分な手法が存在するとは言いがたく、テスト支援ツールについても同様である。そこで本論文では、並行処理プログラムのためのテストを支援するツールに関する考察を行う。

## 2. テストの手順

プログラムのテスト(逐次処理・並行処理の両方において)は、一般に次のような手順・方法で行なわれている。

## i) テストケースの作成

機能や構造に基づいて、プログラムの入力についての条件(入力条件)、およびプログラムが正しい場合の出力(期待結果)をテストケースとして作成する。

## ii) テストデータの選定

テストケースの入力条件を満たす入力データ(テストデータ)を選定する。

## iii) テストの実施

テストデータを用いて、被テストプログラムを実行する。

## iv) 実行結果の判定

テストデータを用いて実行した被テストプログラムの出力を、期待結果と比較する。

## v) テストの評価

テストの状況を分析し、必要であればテストデータの追加を行なう。

テストの目的は、プログラム内の誤りを見つけること、および、プログラムの正しさに対する確信を増すことである。しかしながら、これは同時にテストの限界で

もある。すなわち、テストにおいては、実際に実行したテストデータに対してのみ、被テストプログラムの正しさを保証できるだけであるので、テストによって、被テストプログラムが正当である(すなわち、誤りが存在しない)ことを証明することは、現実的に不可能である。そこで、手順 v) において、テストをどこで終了とするかが問題となる。

テストの終了条件を定めたものをテスト基準と呼ぶ<sup>[1]</sup>。並行処理プログラムのためのテスト基準については、以前からいろいろ議論されてきた<sup>[2]</sup>。従来、テスト基準は、手順 v) において適用していた。テストを行なう際に、テスト基準を最初に設定すれば、その基準を満足するようなテストケースを作成することができる。こうして作成したテストケースに基づいて選定したテストデータを用いてテストを行なうことができれば、テストは充分である(テスト基準を満足するテストを行なった)と判断できるので、テスト作業をその時点で終了とすることができる。

今回は、並行処理プログラムについてテスト基準を先に定め、その後、前述の手順にしたがって、テスト作業を行なう場合のツールについて考察する。

## 3. ツールの概要

ツールの概要を図 1 に示す。ツールの機能としては、テストケース作成部・強制実行用前処理部・テスト充分性評価部、および、各工程で使用するデータを保持するためのテスト用データベースからなっている。以下、それぞれの機能について見ていく。ただし、テスト基準は既に定めてあるものとする。

## 3.1 テストケース作成部

被テストプログラムを入力とし、その内部構造に着目して、与えられたテスト基準を満足するように、テストケースとして、協調路<sup>[3][4]</sup>を出力とする(手順 i))。協調路は、プログラム内のそれぞれのタスクから、同期関係を満足するように路を取り出して組にしたものであり、並行事象の半順序関係を保持する。

## 3.2 強制実行用前処理部

並行処理プログラムの特徴として、非決定性が挙げられる。これは、同じテストデータを与えても、その時の状態や環境によってプログラムが異なる動作をすることである。そのために、テストデータを与えただけでは、並行処理プログラムが協調路として与えられたテストケースの通りに実行される保障がない。そこで、協調路

A Study of a Testing Support Tool for Concurrent Programs.

Tetsuro KATAYAMA\*, Toshiyuki KOMODA\*, Zengo FURUKAWA\*\* and Kazuo USHIJIMA\*

\*Department of Computer Science and Communication Engineering, Kyushu University.

\*\*Educational Center for Information Processing, Kyushu University.

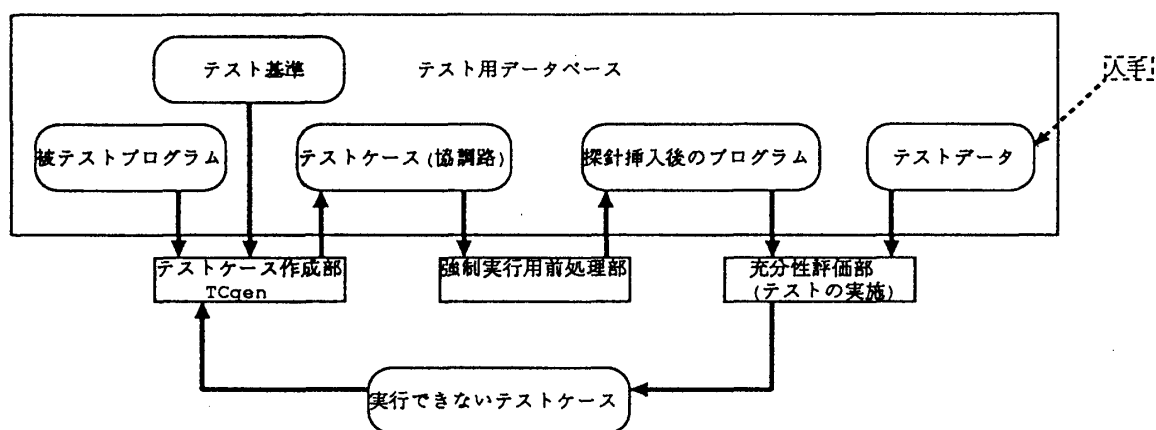


図 1: ツールの概要

の通りに並行処理プログラムを実行するために、並行処理プログラムの強制実行が必要となる。強制実行を行なうために、被テストプログラムに探針を挿入する方法を考えた。入力は協調路であり、出力は、入力に応じて探針を挿入されたプログラムである。

### 3.3 テスト充分性評価部

被テストプログラムに対し、テストデータを選定する(手順 ii))。そのテストデータを用いて、実際にテストを実施する(手順 iii))。探針挿入後の被テストプログラムをすべて実行できれば、テスト基準は満足したことになるので、テスト作業を終了できる。

### 3.4 テスト用データベース

今まで述べてきたように、テスト作業に必要なデータとして、被テストプログラム・テストケース・テストデータ・テスト基準・探針挿入後の被テストプログラムがある。これらのデータを一括管理・即時利用するために、テスト用データベースが必要となる。

## 4. 考察および検討課題

テストケース作成部については、現在までに、プログラミング言語 Ada で記述された並行処理プログラムを入力とし、テスト基準として、枝被覆基準、ループ被覆基準、相互作用被覆基準を満足するような協調路を出力とするテストケース生成ツール TCgen を試作した<sup>[4]</sup>。

強制実行前処理部については、挿入する探針の種類や、実際の挿入方法について考察を行なった<sup>[5]</sup>。また同時に、探針を挿入することによる、被テストプログラムに与える影響をも考慮しなければならない。

テスト充分性評価部については以下に述べるような検討課題がある。

- i) テストデータの生成において、テストデータを自動的に生成するには、被テストプログラムに存在する条件式の不等式を解く必要がある。これは、一般に非可解であるので、テストデータ自動生成のツール

化は不可能である。テストデータは、被テストプログラムを参照して、人手により与える必要がある。

- ii) 今回の方法は、テスト基準を最初に設定することにより、テストケースとしての協調路を機械的に作成する。このため、テストデータが存在しないような協調路、すなわち、実行不可能なテストケースを作成する可能性がある。実行不可能なテストケースが存在した場合、与えられたテスト基準に対して、被テストプログラム中のどの部分が実行できないかを探索し、その部分の実行を実現するテストケースを追加する必要がある。

テスト用データベースについては、現在まだ詳細な検討が済んでいないので、さらに詰めていく予定である。また、テスト用データベースは、プログラムの保守にも役立つことが期待できる。

## 5. おわりに

今回は、並行処理プログラムのためのテスト支援ツールについて検討した。今後の課題としては、ツールの詳細化・具体化が挙げられる。

## 参考文献

- [1] 玉井哲雄, 三嶋良武, 松田茂広: 「ソフトウェアのテスト技法」, 共立出版 (1988).
- [2] 古川善吾, 牛島和夫: “並行処理プログラムのテスト法に関する一考察,” 日本ソフトウェア科学会第6回大会論文集, pp.185-188 (1989).
- [3] Katayama, T., Furukawa, Z. and Ushijima, K.: “Event-Constraint Model of a Concurrent Program for Test-Case Generation,” *Proc. JCSE'92*, pp.285-292 (1992).
- [4] 片山徹郎, 菰田敏行, 古川善吾, 牛島和夫: “並行処理プログラムのためのテストケース生成系の試作,” 情報処理学会研究報告, Vol.92, No.59, pp.9-16 (1992).
- [5] 菰田敏行, 片山徹郎, 古川善吾, 牛島和夫: “Ada 並行処理プログラムのテストケース作成とその強制実行に関する一考察,” 第20回 JAPAN SIGAda 予稿集, pp.9-15 (1993).