

4R-11

分散マルチプロセスのテストにおけるイベント系列制御方式

橋本 辰範 生沼 守英 堀田 博文

NTTソフトウェア研究所

1 はじめに

一般に、分散システム全体の動作は、そのシステム下で並行に動作している各プロセスの動作およびその順序に左右される。分散システムでは一般にプロセスの動作順序に非決定性が存在するため、この非決定性をもつ部分に関わる設計と実現が正しくなされているかどうかをテストで確認する必要がある。このテストを行うためには、非決定的な部分を自在に制御する機構が不可欠となる。

本論文では、分散システムのテストを目的とし、その非決定的な部分を制御する方式について提案する。

2 分散マルチプロセス

ネットワークで結合された複数のコンピュータに分散配置されたプロセスが協調動作しながらある目的を達成しようとする形態を想定する。この形態を分散マルチプロセスと定義する(図1)。本稿で扱う分散マルチプロセスモデルは、プロセス間のインタフェースが以下に示すメッセージ送受信だけで行われるという制約をもつものとする。

・ send (in 送信元, in 送信先, in メッセージ)

送信元から送信先にメッセージを送り走行し続ける。

・ receive (out 送信元, out メッセージ, in 待ち時間)

自分に送信されたメッセージがある場合、その送信元とメッセージを返却する。メッセージが未着の場合、待ち時間を限度に待ちにはいる。待ち時間経過後は、タイムアウト(メッセージ受信不可であること)を返却し走行を再開する。

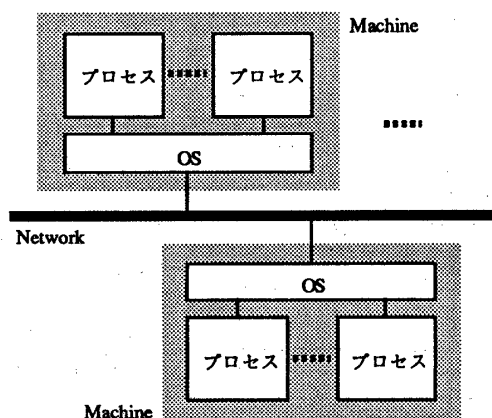


図1 分散マルチプロセスの形態

一般に、分散マルチプロセスシステムにおいて非決定性の生じる原因には次のものがある。

- ①通信の遅延
- ②プロセスのスケジューリング
- ③共有変数へのアクセスタイミング
- ④タイマ割込み
- ⑤外部割込み
- ⑥時刻の値を利用した処理

ただし、本稿でのモデルでは、前述の制約により③に起因する非決定性は生じない。なお、④⑤に起因する非決定性については本稿では触れない。このモデルの非決定性については①と②に関しては通信の順序のばらつき、そして⑥に関しては時刻の値のばらつきという現象だけで現れる。よって、この非決定性に関するプロセスの動作を示すイベント系列は以下のイベントにより構成することができる。

- send (送信元, 送信先, メッセージ) ---送信
- receive (送信元, メッセージ, 待ち時間) ---受信
- time (時刻) ---時刻の参照

3 イベント系列の制御

3.1 イベント制御エージェント

分散マルチプロセスモデルにおける任意のプロセスのテストにおいて、前もって用意されたイベント系列に沿った動作を行わせることを目的にイベント制御エージェントを導入する。イベント制御エージェントは、テストの対象となる各プロセスに一对一に張り付けてメッセージ送受信を代行することにより通信の順序を制御する(図2)機能をもつ。テスト対象として選ばれたプロセス群をテスト空間と定義すると、各イベント制御エージェントの動作は以下の3つに大別される。

(1) テスト空間内に閉じた信号に対する動作

- ・被テストプロセス以外からのメッセージ受信

即座に受信しその内容(送信元とメッセージの組)を受信バッファに追加する。受信が完了した後は、再び受信待ちとする。
- ・被テストプロセスからのメッセージ送信要求

そのまま送信する。
- ・被テストプロセスからのメッセージ受信要求

設定されたイベント系列がタイムアウトのときは、それを被テストプロセスに引き渡す。それ以外の場合は受信バッファを調べ、設定されたイベント系列に適合した信号があればその内容を被テストプロセスに引き渡す。適合した信号がなければ受信バッファに新たな信号が追加されるまで待ち、被テストプログラムをブロックする。

- (2) テスト空間外との信号に対する動作
 - ・被テストプロセス以外からのメッセージ受信
即座に受信はするがその内容は受信バッファには追加しない。受信が完了した後は、再び受信待ちとする。
 - ・被テストプロセスからのメッセージ送信要求
なにもせずに正常終了したように振舞う。
 - ・被テストプロセスからのメッセージ受信要求
設定されたイベント系列に適合する信号内容を被テストプロセスに引き渡す。

(3) 時刻の参照

イベント系列で用意された値を返す。
イベント制御エージェントは被テストプロセスの数だけ存在する。また、制御すべきプロセスのイベント系列は各イベント制御エージェントがローカルに管理する。

3.2 イベント系列のテスト

以下の工程に従い分散マルチプロセスシステムモデルに起こり得るイベント系列をテストする。

(1) イベント系列とテストデータの作成

まず、プロセスの仕様から論理的に導き出すことができるテストデータ、すなわち受信メッセージの内容や時刻間い合わせに対する値を作成する。そして、このテストデータから予測されるプロセスの動作の中から非決定的な部分を抽出し、イベント系列を作成する。

(2) プロセス単体テスト

被テストプロセスに対し用意されたすべてのテストデータ、イベント系列についてテストする。メッセージ送受はすべてイベント制御エージェント内で解決され、他プロセスとのメッセージ送受は実際には行われない。

(3) プロセス結合テスト

複数の被テストプロセスを同時に動作させテストする。

被テストプロセス間に閉じたメッセージは実際に送受されるが、その受け取りタイミングだけがイベント制御エージェントにより制御される。

3.3 本方式による利点

本方式は、分散システムを構成する各プロセスの制御をイベント制御エージェントという形態により、分散、協調型で実施するという特徴をもつ。この結果、イベント系列制御に関する負荷を分散することができ、さらに、テスト実施時においてもプロセスの並列性を残すことができる。

同様の特徴をもつイベント系列制御方式は、メールボックスの制御による通信ソフトウェアの再現テスト[1]でも見られる。しかし、本方式のように個々のプロセスに着目したテストの実施については考慮されていない。本方式におけるイベント制御エージェントは、イベント系列の制御だけでなく、通信のグミー化の機構をもつことにより、プロセス単体テストやプロセス結合テストといった分散システムの段階的テストを可能とする。

4 おわりに

本稿では、分散マルチプロセスシステムを対象に、イベント系列の制御を行うイベント制御エージェントの機構を導入し、システムの非決定性に起因するバグをテストで発見する方式を提案した。今後、イベント系列とテストデータの作成方式や、割込みに起因する非決定性の制御について検討を加え、本方式との融合を図る。

参考文献

[1] Tai,K. and Ahuja,S., "Reproducible Testing of Communication Software", Proc. Compsac87, 331-337.

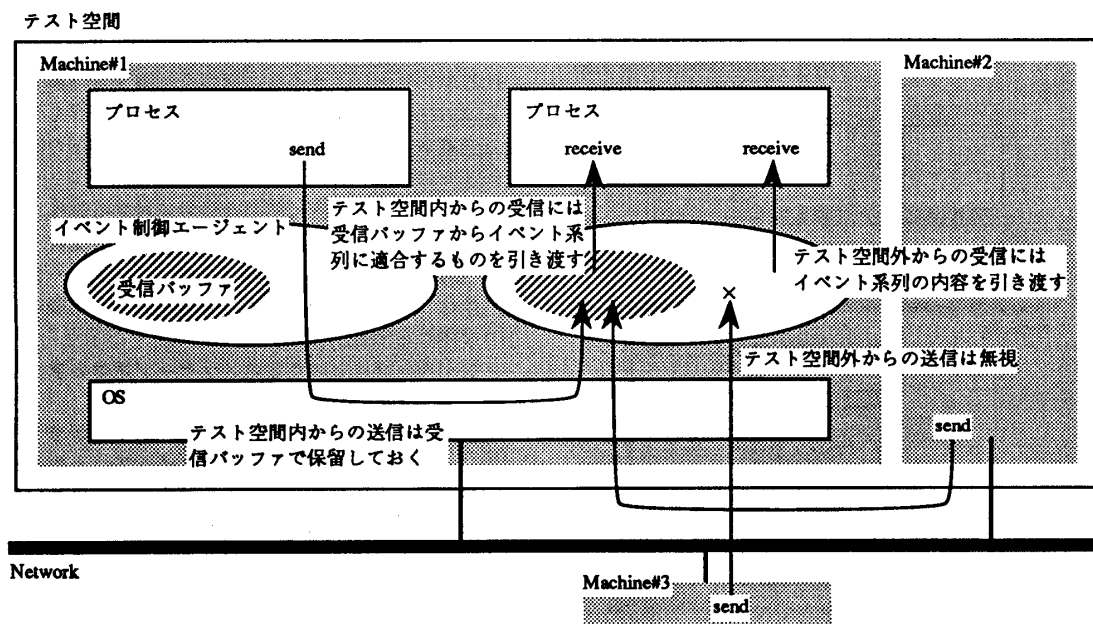


図2 通信順序の制御