

## 5 A E - 9 離散系シミュレータをシステムシミュレーションテストへ 適用するための時刻同期手法の開発

○石井 良和、 山田 直之、 中野利彦\*、 東原敏昭\*、 藤岡誠一\*

日立製作所 電力電機開発本部、日立製作所 大みか工場\*

### 1. はじめに

工場の製造ラインの運転方案検討や生産能力評価、情報システムの通信手順検討や性能評価など、システムの上流設計段階において重要な役割を果たしている離散系のシミュレータを、製作した制御装置のシステムテストにも適用するための要素技術を開発している。

### 2. 離散系シミュレータの課題

離散系のシミュレータではモデル上の時刻(以下、モデル時刻)の進展が、シミュレーション内容などに応じて不規則になり、制御装置からの制御出力が正しいタイミングでシミュレーションに反映できない。そのため制御装置を取り巻く各種設備やオペレータの動作を、離散系のシミュレータで模擬して制御装置のシステムテストを行うのは困難である。本発表では市販のこのような離散系シミュレータのモデル時刻と、実世界の時刻(以下、実時刻)をユーザーが実施できるモデル定義の範囲で同期させ、システムテストにも適用できるようにする方法について述べる。

### 3. モデル上の時刻と実世界の時刻を同期させる方法

本手法は次の4つの条件を満たす離散系シミュレーションソフトにおいて、ユーザーのモデル作成機能の範囲で使用できる汎用的な手法である。

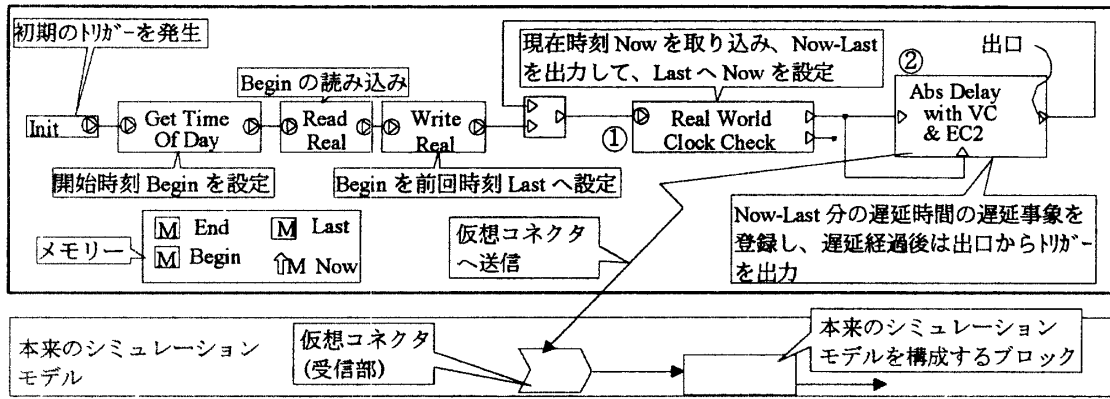
- (1) 予め定義された部品を用い、部品間の関係を記述することでモデルを構築する。
- (2) ユーザーが独自の部品を定義できる。
- (3) 計算機のシステムクロックなどの時刻を取得する関数が、部品定義において使用できる。
- (4) 可変の遅延時間を設定できる時間遅延部品が定義されている(または、定義できる)。

本手法は計算機のクロックを読み込んでモデル時刻とする処理ループを、元々のモデルに追加することで、モデル時刻と実時刻を簡単に同期させるだけでなく、クロック読み込み時に外部入力の処理を行わせることで、外部からの制御入力を正しいタイミングでシミュレーションに反映させる。

図1にこのような処理を行うモデルを示す。図2にモデル時刻と実時刻の関係を示す。時刻計測要素①へはシミュレーション開始と共にトリガーが到着し、計算機のクロックを用いて取得した実時刻からシミュレーション開始からの経過時間  $x_1$  を計算して遅延要素②に渡す。②では  $x_1$  時間後に出口からのトリガー出力事象を設定するが、この  $x_1$  時間の中に本来のモデル上で事象が起きない場

Time synchronization method of discrete event simulator for use of system simulation test.

Ishii Yoshikazu, Yamada Naoyuki, Nakano Toshihiko\*, Higashihara Toshiaki\*, Fujioka Seiichi\*  
Power & Industrial Systems R & D Division, Hitachi, Ltd., \*Omika Works Hitachi, Ltd.



このモデルは BONEs(Block Oriented Network Simulator:Alta Group of Cadence Design Systems, Inc.の登録商標)を用いて構築

図1 モデル上の時刻を実時間に同期させるブロック

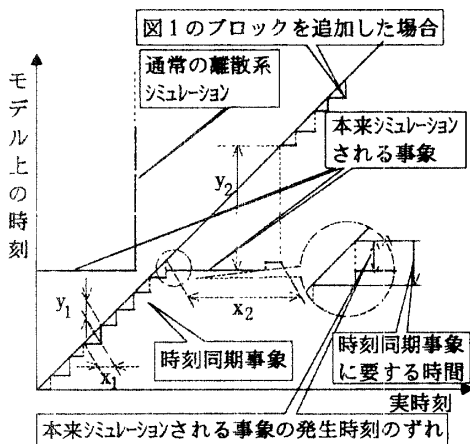


図2 時刻同期事象とその他の事象の処理タイミング

合には、直ちにモデル時刻にこの遅延時間( $=x_1$ )が加算されトリガーが生成される。このトリガーは再び①に到着し、前述の処理が繰り返され、モデル時刻と実時刻が同期する( $\sum y_i = \sum x_i$ となる)。本来のモデル上で生起する事象の方がこのような時刻同期事象よりも早く生起する場合は、その事象が選択され、対応した処理が行われる。この処理に要した時間  $x_1$  は、次の時刻同期事象の遅延時刻  $y_2$  となるため、他の早い事象がなければ、直ちにこの事象が生起して、モデル上の時刻が実時刻に同期する。しかし図3のように、モデル上で連続的に事象が生起し、 $T/\nu > 1$  となる場合には、

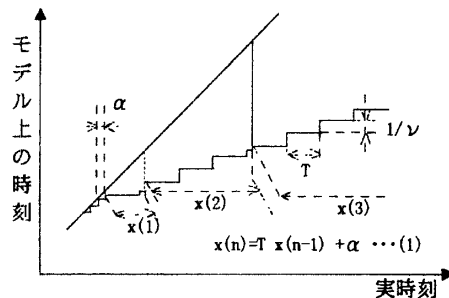
時刻同期の時間幅（時刻同期事象の生起間隔）は図3の式(2)のように発散する。このため本方式では、例えば連続的な状態変化をするプロセスを含む現象をシミュレーションする場合に、時定数に応じてプロセス毎に状態の更新(積分)処理を行って、定期的に生起する事象の平均処理時間を短くするなど、モデル化の時点で工夫が必要である。

更に②の出口からのトリガー出力直前に、外部からの入力の有無をチェックし、有った場合はトリガー出力前に外部入力を処理することにより、制御用計算機などからの入力を時刻同期に要する時間の精度でシミュレーションに反映できる。

#### 4. 結果

1秒当たり最大40個の部品を製造して搬送し、所定の個数にまとめて梱包する製造ライン(4秒毎の生産状況通知も行う)

のシミュレータ(BONEsの部品で  $2 \times 10^3$  個程度の部品からなる。)においては、装置の起動や減速などの制御指令を最大70ms(Hitachi9000/80i 主記憶64Mを使用)程度の誤差でシミュレーションに反映できることを確認し、これにより本手法の有効性を確認した。



T: 本来シミュレーションする事象の平均処理時間  
 $\nu$ : 本来シミュレーションする事象の平均発生間隔  
 $\alpha$ : 時刻同期事象の処理時間(通常  $T > \alpha$ )  
 $x(i)$ : 同期事象の設定する時間=同期事象の生起間隔  
 $x(n) = T(T/\nu)^{n-1} \dots (1)$  (1)で $\alpha$ とする)  $\dots (2)$

図3 時刻同期事象の時間幅