

性能評価シミュレータにおけるトランザクションの発生について

6D-5

西岡 善彦 畑野 宏紀
NTTコミュニケーションウェア株式会社

1. はじめに

システム開発においてその性能を事前に予測するため PC や WS によるシミュレータを用いている。現状の OLTP システムをモデル化したシミュレーションにおいてトランザクションの発生は、当該システムの実運用で想定する数だけ必ず発生し、かつ発生タイミングはランダムとさせたい。これらの条件が同時に満足させられないと、システムに与えるトータル負荷が小さくなりシステム性能を過大評価してしまう可能性がある。

しかし、現在私達が使用している市販のシミュレーション・ツールでは、2つの条件を満足させる為に、トランザクション発生の為の確立分布を駆使し、非常に不便である。そのため、2つの条件「想定したトランザクション数は必ず発生し、かつその発生タイミングはランダムである」を満たし、かつ簡単にトランザクション発生タイミングを導き出すアルゴリズムを考案し、使用しているシミュレータに適用することで、本アルゴリズムの効果を確認した。

2. トランザクション発生アルゴリズムの検討

トランザクション発生タイミングにおける、前提条件を示す。

1. 規定のシミュレーション時間中に規定回数のトランザクションを必ず発生する。
2. トランザクションの発生タイミングはランダムとなる。

上記の条件を満たす発生タイミングを算出するアルゴリズムを示す。

(1) 基本条件を以下のように定義する。

- ① T_{max} : シミュレーション実行時間(秒)
- ② N_{max} : 規定トランザクション数

(2) n回目のトランザクションを実行したときのパラメータを以下のように定義する。

- ① T_{rest_n} : シミュレーションの残り時間(秒)
 - ② N_{rest_n} : トランザクションの残り回数
 - ③ M_{-n} : 前トランザクションの発生時刻を基準とした、本トランザクションの発生タイミング(秒)
 - ④ M_{base_n} : M_{-n} を決定するための「基準値」
- (3) n回目のトランザクションの発生タイミング M_{-n} を決定する手順を下記に示す。

n 回目のトランザクションの発生タイミングを決定するときの

① シミュレーションの残り実行時間 T_{rest_n} と残りトランザクション数 N_{rest_n} を求める。

$$\begin{cases} T_{rest_n} = T_{rest_(n-1)} - M_{-(n-1)} & (n \geq 2) \\ \quad \quad \quad = \text{前回の残り時間} - \text{前回のトランザクションの発生タイミング} \\ T_{rest_1} = T_{max} & (n = 1) \end{cases}$$

$$\begin{cases} N_{rest_n} = N_{rest_(n-1)} - 1 & (n \geq 2) \\ N_{rest_1} = N_{max} & (n = 1) \end{cases}$$

② 発生タイミングの「基準値」を求める。

$$M_{base_n} = \frac{T_{rest_n}}{N_{rest_n} + 1} = \frac{\text{残り時間}}{\text{残トランザクション数} + 1}$$

③ M_{-n} の発生タイミングが取りうる値の範囲を、②より以下の範囲とし、この内から1点を等しい確率で選択する。

$$0 \leq M_{-n} \leq 2 \times M_{base_n}$$

④ $N_{rest_n} = 0$ でなければ①～③を繰り返す。

3. アルゴリズムの証明

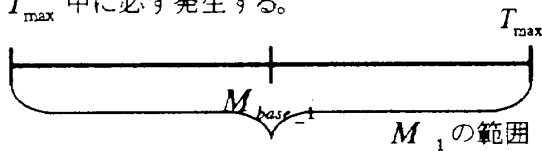
考案したアルゴリズムが、規定時間内に全て発生可能であることを、以下に証明する。

2. (3). ③の仮定より、 M_n が取りうる最大値を $2 \times M_n$ とした。展開すると、

$$2M_n = \frac{2}{N_{rest_n} + 1} T_{rest_n} \dots (a)$$

① $N_{max} = 1$ のとき、 $N_{rest_1} = 1$ となるため、

$2M_1 = T_{rest_1} = T_{max}$ となる。よって M_1 は T_{max} 中に必ず発生する。

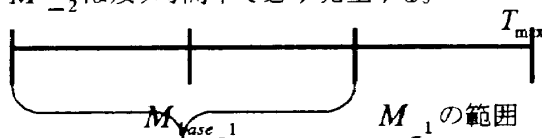


② $N_{max} = 2$ のとき、 $N_{rest_1} = 2$ となるため、

$$2M_1 = \frac{2}{3} T_{rest_1} = \frac{2}{3} T_{max}$$

となり、少なくとも残り時間が $T_{max}/3$ 存在する。よって①と同様に

M_{-2} は残り時間中で必ず発生する。



③ $N_{max} = n$ のとき、

$$2M_n = 2 \times \frac{1}{N_{rest_n} + 1} T_{rest_n} \leq T_{rest_n} \text{ (等号条件は } N_{rest_n} = 1 \text{)}$$

よって、②より $N_{rest} > 1$ の場合、必ず残り時間が発生する。また①より、 $N_{rest} = 1$ の場合だけ、残り時間以内の値を取る。よって、規定時間内に全てのトランザクションが発生する。

4. アルゴリズムの妥当性検証

考案したアルゴリズムが前提条件を満たすことを検証する為、通常、私たちが行っている性能評価シミュレーションにおける一般的なパターンを条件として用いて確認した。

(1) シミュレーション実行時間 (T_{max}) 300, 30 (秒)

(2) 規定トランザクション数 (N_{max})

{10, 50, 100, 300, 600}

このうち、 $(T_{max}, N_{max}) = (30, 30)$ で複数回施行

した結果を以下に示す。

図4-1は、シミュレーション時間 T_{max} (秒) における各トランザクションの発生タイミングのパターン分布を表わしている。また図4-2は、各トランザクションの発生タイミングの分布図である。

これにより、トランザクションの発生がランダムかつ、シミュレーション時間内に収まることを確認した。

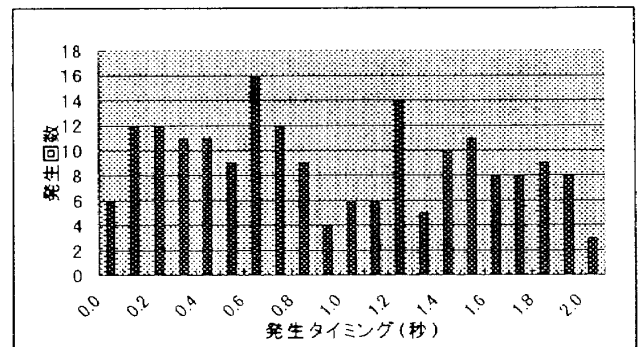
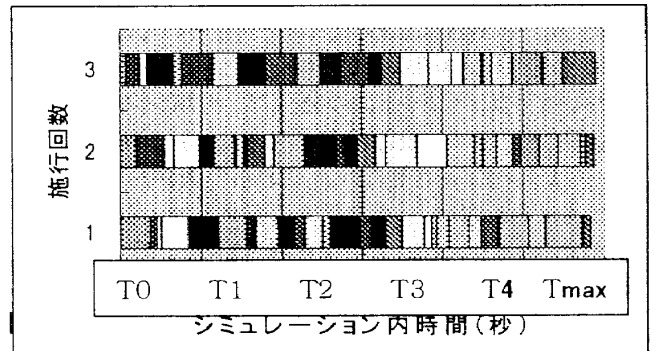


図4-2 算出した発生タイミングの分布図

5. シミュレーションへの適用

私達が使用するシミュレータには、トランザクション情報を、市販のトラフィック情報アナライザから外部入力により利用することが可能である。

そこで、2. で算出したトランザクション発生タイミングを、トラフィック情報アナライザのデータ形式に変換することにより、シミュレータに適用した。

6. まとめ

本論文のアルゴリズムにより、シミュレーション時間内における規定トランザクション数の発生が可能となり、より正確な実態に近いシミュレーションが可能となった。

参考文献

- [1] 亀田壽夫・紀一誠・李著：性能評価の基礎と応用，共立出版，1998
- [2] OR編集部会：初等ORテキスト，日科技連出版，1989