

大規模組込みシステムの ハードウェアリソースシミュレータの開発

桜井 祐市[†] 新保 健一[†] 鳥羽 忠信[†]
株式会社 日立製作所 横浜研究所[†]

1. はじめに

コンピュータ等制御機器を搭載した製造装置や分析、解析装置などに代表される産業用の大規模組込みリアルタイムデータ処理装置において、一時的なネットワーク負荷増大によるバッファオーバーフローはシステム停止を引き起こす可能性がある。装置の安定稼働のためにはハードウェアリソースを事前に高精度に予測することが重要である。しかし、近年大容量データの高速処理要求の高まりにより装置が大規模化しており、ハードウェアリソースの予測が困難になりつつあることから、高精度かつ高速なシステムシミュレーション技術が求められている。

2. ネットワーク性能シミュレータ HAPSIM

この背景の下で、図 1 に示すネットワーク性能シミュレータ HAPSIM (Hardware Performance SIMulator) を開発した。HAPSIM は大規模組込み処理装置のハードウェアリソース評価を目的とする MATLAB/Simulink ベースのネットワーク性能シミュレータである。システムの安定稼働を目標にネットワークのバッファ容量などのハードウェアリソースを事前に予測する。ハードウェアの挙動をモデリングするシミュレーション言語としては HDL (Hardware Description Language, ハードウェア記述言語) が多く用いられているが、システムレベルのバッファ使用量評価が実用的な計算時間に収まらない課題が存在した。これに対し HAPSIM はシステムのバッファ使用量変動要因を単純化し、処理の高速化を図った。ネットワークトポロジに起因するバッファ使用量変動については、パケットレイテンシをハードウェアから事前に抽出しネットワークスイッチモデルに応用することで、HDL に比べレイテンシ算出処理を簡略化した。また CPU コア処理起因のバッファ使用量変動については、プロセッサを 2 状態の共有バス時間モデルで離散的に再現し計算処理の簡略化を図った。HAPSIM はバッファ使用量の一時的な使用量増大など定常状態以外の状態を観測すべく、パケット遷移を基準とした Discrete-Event Network Simulator として開発した。パケット遷移の離散シミュレーションにより、高速計算と過渡状態把握の両立を実現し、輻輳発生を事前に確認し、必要なバッファ量を予測可能である。開発した HAPSIM を用いた実験では、HDL の結果との比較を主眼とするプロセッサエレメント負荷に起因するバッファ使用量増大のみをモデリングする実験にて、HDL によるハードウェアモデルと比較した結果、実行時間を約 1/12 (誤差:1064Bytes, 4 パケット分) に短縮した[1]。

HAPSIM はギガオーダネットワークの高精度シミュレーションを目的としており、MATLAB サンプル時間を 1ns に

設定した。MATLAB では C 言語で開発したシミュレーションモジュール (MEX:MATLAB EXecutable) はサンプル時間間隔で動作する。HAPSIM も MEX ファイルでハードの挙動をシミュレーションするモジュールにおいては、1ns 間隔で状態更新を行っており、本手法によりビットデータレベルでの変化の高精度なシミュレーションを実現している。しかし、本来ならパケット遷移イベント毎に処理可能なもの、例えばスイッチバッファへのパケット格納処理も、常に 1ns 毎の状態更新計算を行っており、本来は不要な計算処理を実行している事がわかった。

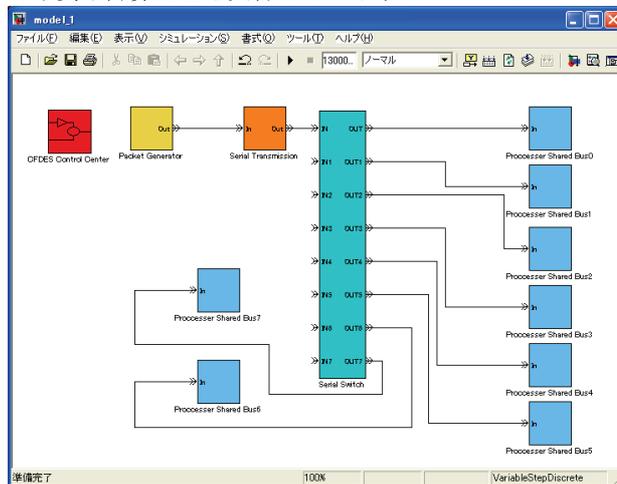


図 1 HAPSIM シミュレーションモデル設計画面

3. タイムホイールベース時間管理

MATLAB は MEX ファイルの状態更新を単位時間 (unit time) ごとに行うため、本来は不要な状態更新計算も全て行いシミュレーション時間が長くなる。高精度が求められる箇所は従来の HAPSIM 同様 1ns 間隔で状態更新計算を行い、それ以外の箇所は求める精度に応じて状態更新計算間隔を可変としたい。よって今回、大規模組込みリアルタイムデータ処理装置の高効率シミュレーションを行うために、MATLAB の時間管理とは独立した仮想のシミュレーションステップ (v_step) として時間管理を行う。タイムホイール (TimeWheel) 管理を採用したマルチレートシミュレーションを開発した。図 2 に処理の概要を、図 3 に MATLAB の時間管理とタイムホイール (v_time) による時間管理との関係を示す。第一に、MATLAB のシミュレーションタイムステップで全てのブロックが動作し、状態更新を行う (①)。次に、全てのブロックにおいて次に起こるイベント駆動リクエストを集約し、イベントとしてタイムホイールに登録する (②)。次の MATLAB シミュレーションタイムステップ (③) にて、仮想シミュレーションステップ (v_time) 上の現シミュレーション時点から、最も早く起きるイベントの時間を得るべく v_time のタイムホイールを進め、 v_time の現在 (current) 時

Development of Hardware Resource Simulator for High-Performance Embedded Systems

[†] Yuichi Sakurai, Kenichi Shimbo, Tadanobu Toba, Yokohama Laboratory, Hitachi, Ltd.

間を進捗させる。この current 時刻において全てのブロックが動作し、状態更新を行う。そして、次に起こるイベント駆動リクエストを集約し、イベントとしてタイムホイールに登録(④)、次の MATLAB シミュレーションタイムステップ(⑤)にて v_time のタイムホイールを進め、v_time の現在 (current) 時間を進捗させる。これら一連の処理を繰り返し実行する事で、マルチレートシミュレーションを実現する。以上のシミュレーション対象モデルの状態に応じた離散時間管理により MATLAB タイムステップ数を最小化する。

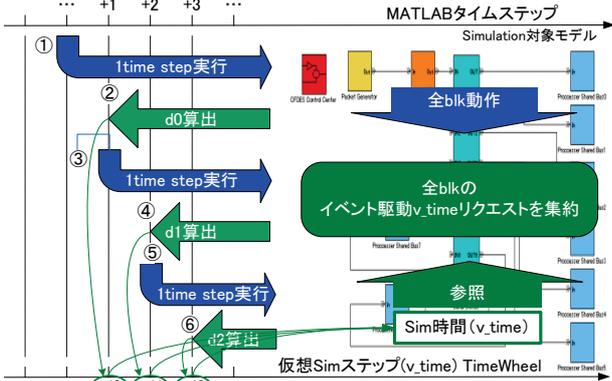


図 2 タイムホイール管理のマルチレート処理

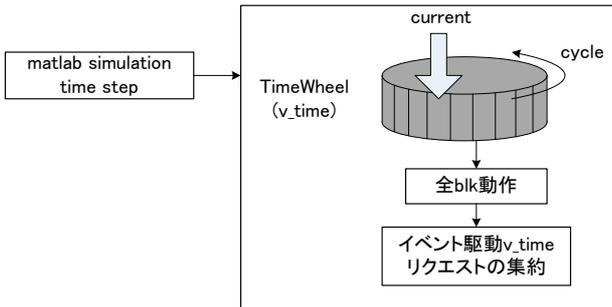


図 3 MATLAB 時間管理とタイムホイール管理の関係

本提案のタイムホイール方式は、MATLAB タイムステップを最少化することになり計算時間が短縮される。しかし、その短縮率は対象とするモデルが必要とするシミュレーション精度によって異なる。そこで、条件を変えたモデルを用いてシミュレーション計算時間を評価し、提案方式の効果を検証する。

4. 方式検証

タイムホイール方式を HAPSIM に実装し、表 1の条件で実験を行った。図 4にモデル構成を示す。ジェネレータからスイッチを介しプロセッサエレメントにデータを一方伝送する。本検証ではジェネレータ数を変化させ、スイッチ内部で故意にパケット衝突を起こす。ジェネレータ数が増えパケット衝突イベントが増えるとポートスイッチ処理が複雑化し、高精度を維持するためにタイムホイールに登録される計算イベント数が増大し、モデルの状態更新間隔が短くなる。この状態更新間隔が可変となることを確認する。ジェネレータ数 1(パケット衝突無し)で本提案の効果が最も表れ、ジェネレータ数が増えると本提案の効果が減少するものと予測した。結果について図 5に示す。ここでグラフは速度向上を比で表し(従来方式速度/提案方式速度)で求めた。ジェネレータ

数が 1 個でパケット衝突が無い場合、提案方式により従来の HAPSIM に比べ 62 倍高速化した。また、衝突数が増えるにつれ速度向上比が減少しており、ジェネレータ数が 8 個の場合は 1.6 倍の速度向上となった。これは、状態更新間隔を細かくし、従来の HAPSIM 同様に高精度な計算を実施しているためである。実際のシステムでは本実験のように一定間隔パケット連続伝送ではなく、伝送容量にはばらつきがあり、例えばパケット衝突が無く高精度計算が不要な状態とネットワーク混雑によるパケット衝突状態とが混在する。それぞれの状態に応じて本提案方式は MATLAB タイムステップ数を最小化し計算時間の短縮を実現する。

表 1 実験条件

#	実験条件	設定
1	ジェネレータ数	1~8
2	プロセッサエレメント数	1
3	伝送パケット量	10,000 パケット
4	伝送パターン	全てのジェネレータから一定間隔パケット連続伝送
5	伝送容量	一定容量(1.5Gbps)伝送

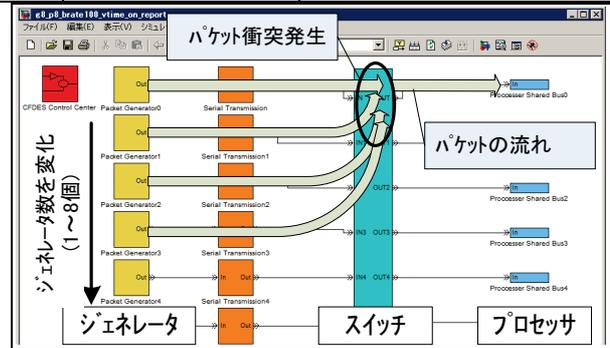


図 4 検証用シミュレーションモデル構成

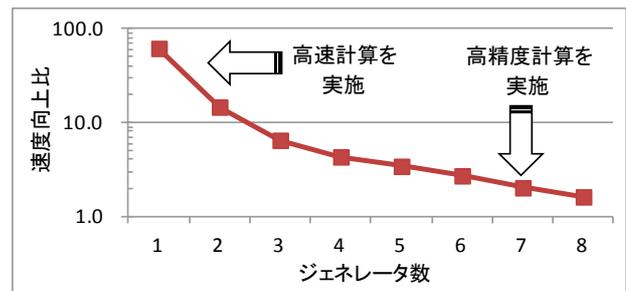


図 5 実験結果

5. まとめ

ネットワーク性能シミュレータの性能向上のため、仮定のシミュレーション時間管理をするタイムホイールベースの離散イベント時間管理を提案した。MATLAB タイムステップと連動する事によりシミュレーション速度が最大約 60 倍向上することを確認した。今後は、実システムのシミュレーションへの適用が課題である。

参考文献

[1] 桜井 祐市, 新保 健一, 鳥羽 忠信, “ネットワーク性能シミュレータの開発とバスのバッファ評価への適用”, 組込みシステムシンポジウム 2012, 2012 年 10 月.