

平成23年度 情報工学コース卒業研究報告要旨

枝廣 研究室	氏 名	鈴木 悠 太
卒業研究題目	マルチコア・メニーコアに向けた Simulink モデル記述法に関する一考察	
<p>近年、組み込みシステムにおいてもマルチコア CPU の急速な普及が進んでいる。しかし、コードレベルでのアプリケーション並列化は一般的に容易ではなく、より上位レベルでのソフトウェア並列化が注目されている。本研究では、車載システムなどの制御システム分野においてデファクトスタンダードとなっている、MATLAB/Simulink モデルベース設計に着目する。開発効率に加えて、モデルには視覚的な構造的並列性を抽出しやすいという利点がある。ハードウェアリソースを有効的に活用するため、マルチコア・メニーコアを意識した Simulink モデルの記述法を確立することは有意義である。しかし、現在の制御モデル設計と組み込み実装の間には深い溝があり、既存のモデルの中には、並列性能を引き出すことが困難なものが多々存在する。今後は、制御設計時に記述が容易で、実装時の性能が期待できるモデル記述を検討していく必要があると考えられる。</p> <p>本研究では、FIR フィルター型の計算モデルを起点に、NEC 社において研究開発中のモデルベース並列化ツールを使用し、マルチコア・メニーコアに向けた Simulink モデル記述に関する考察を行った。ツールを併用することで、モデル記述上にてコードレベルにおけるタスク間の依存関係や粒度を調整することが可能となる。</p> <p>ハードウェアのリソース制約や、通信オーバーヘッドの削減などの観点から、タスク数、タスク間通信回数を削減することは重要である。そのためには、モデルを構成する複数のブロックを1つのタスクにマッピングしなければならないが、その際、並列性に注意する必要がある。一例として、アルゴリズムを表す Z 変換の式変形からタスク分割を行う既存手法を、Simulink モデルに適用した。システムの入出力に対して、ブロードキャストまたはパイプラインを設定した複数のモデル設計に加えて、構造的並列性を意識した fork-join 構造設計に関する検討、評価を行った。各ブロックをタスクにマッピングしない場合と比較して、より良いタスク設計が存在することが確認できた。また、モデルベース並列化による性能は、構造的並列性に加えて、アルゴリズムを表す初期ブロック配置に大きく依存し、加えて各タスク間のロードバランスにも注意が必要であると考えられる。</p> <p>現実的なモデルの場合、各ブロック及び機能単位であるサブシステム間には計算粒度の不均衡があり、これはしばしばタスク並列パイプライン実行の際のボトルネックとなる。本研究では評価用モデルのブロック粒度をランダムに変更し、並列性を落とすことなくボトルネックの生成を回避する提案手法の評価を行った。複数ブロックをマッピングしたタスクの入出力に着目し、並列性に影響を与えるブロードキャスト先タスク間の依存関係が生じないように粒度を調整した結果、人手による工夫を行わない場合、既存の設計手法の2通りと比較して、並列性を向上させることが確認できた。</p> <p>但し、粒度調整に伴いタスクあたりの通信回数が増加する点は今後の課題であり、実装アーキテクチャも含めた調整が要ると考えられる。また、より現実的なモデルに対しても、マルチコアに向けた種々の記述法を検討することが必要である。このような細かい粒度調整を行うためには、サブシステムの階層を展開した上で、モデル全体のブロックを考慮することが有意義であると考えられる。しかし、現在の商用ツール・制御モデルには様々な課題・制約が存在し、性能向上に加えて開発効率の点でも、今後の研究が必要とされる部分であると言える。</p>		