

平成28年度 情報工学コース卒業研究報告要旨

枝廣 研究室	氏 名	波多野 友貴
卒業研究題目	LLVMを用いたC言語構造抽出と並列実行時誤作動検出	
<p>近年、メニーコアおよびマルチコアプロセッサの重要性が高まっている。複雑で高度なデータを処理するのに、単一プロセッサの性能を上げて、効率的な処理が期待できない時代になっている。そのため、情報処理をするプロセッサの数を増やすことで、この問題を解決する必要がある。しかし、従来のソフトウェア開発はシングルプロセッサ向けのプログラミング開発であったため、簡単にマルチコアおよびメニーコアプロセッサに対応できない。そのため、マルチコアおよびメニーコア向けソフトウェアの開発手法の研究を行い、今後のメニーコアおよびマルチコアプロセッサの性能要求に応える必要がある。</p> <p>枝廣研究室では、車載制御設計等で用いられるモデルベース開発への適用を目指して、モデルベース並列化手法を研究し、自動でタスクをコアに割り当てる手法を研究している。ここでは、Simulinkモデルからブロック構造と性能情報を抽出し、BLXMLと呼ぶXML記述で表現している。また、モデルベース並列化はモデルレベルでのマルチコア向け設計を可能にし、設計を上流にシフトさせ、生産性を向上させる可能性がある。しかし、現在の組み込みシステムの主要ベースであるC言語では、制御構造が複雑で命令表現が多様である。さらにその既存資産（レガシコード）が長期間残り、モデルとCコードが共存する時期が今後長く続くため、共存できる設計環境が必要である。そのため、制御構造や命令表現を扱いやすくするためのブロック構造に分割する必要がある、C言語コードをBLXMLの中で表現しなければならない。</p> <p>そのため本研究では、C言語からBLXMLへの変換の方法を提案し、簡単な例での手動変換をすることで、現状のC言語の構造をLLVMを用いて制御構造や命令表現を扱いやすくする。その目的のために、[ブロック線図での変換]、[関数レベルでの変換]、[ベーシックブロックレベルでの変換]、[命令レベルでの変換]といった、4つの手法を提案し、それぞれの手法に対して評価を与えた。[ブロック線図での変換]ではC言語を、ブロック線図を通して、BLXMLへ変換する方法を述べる。他の3つの手法はLLVMIRを通して、BLXMLへの変換を行う。[関数レベルでの変換]では、C言語関数の引数と返り値に着目して変換を行う。[ベーシックブロックレベルでの変換]では、制御フローとデータフローに着目する。[命令レベルでの変換]では、LLVMIR内のcall関数に着目して、変換を行う。</p> <p>また、本研究では、構造化されたC言語コードの利用例として、構造化された情報を用い、並列化によって実行順序が変化することによる問題について検討した。構造解析を行うため、C言語からブロック抽出およびコード抽出を行う。そのためにまず簡単なC言語コードの並列プログラムを用意し、LLVMを用いて、それを扱いやすい中間コードであるLLVMIRに変換する。その後LLVMIRから実行サイクル数の見積りを生成する。次に見積りを踏まえた動作確認を行った後、誤動作の発生条件を確認する。最後に確認した誤動作発生条件を下に、並列実行時の誤動作を再現する。</p> <p>今後は、C言語からBLXMLへの変換の方法を一般化し、自動で行うツールの開発が期待される。</p>		