氏名	砥崎賢一（かきざきけんいち）
博士の専攻分野の名称	博士（工学）
学位記番号	第10569号
学位授与年月日	平成5年3月16日
学位授与の要件	学位規則第4条第2項該当
学位論文名	PROLOG処理系の実現方式に関する研究
論文審査委員	（主査）教授 豊田 順一 （副査）教授 首藤 勝　教授 北橋 忠宏　教授 溝口理一郎

論文内容の要旨

本論文は，PROLOG処理系の処理速度，メモリ効率，表現能力を向上させることに主眼をおき，最近広く利用されているRISC（Reduced Instruction Set Computer）への実装を前提とした実現方式に関する研究内容をまとめたものである。提案方式は，(1)メモリアクセス頻度を低減する，(2)分岐処理を低減する，(3)パイプラインの空きを有効に利用するなどの，RISCアーキテクチャの高速性を効果的に引き出す実現方式となっている。

第1章では，本研究の目的および意義について述べ，第2章では，従来のPROLOG処理系に関する研究について概説した。

第3章では，PROLOGコンパイラの最適化方式について述べた。最適化方式としては，(1)PROLOG処理系の基本的な実現方式であるWAM（Warren Abstract Machine）の抽象命令の特殊化と複合化による最適化，(2)出力モードの単一化処理の最適化，(3)非決定性処理の除去による最適化の各方式を提案した。提案方式のappend／3述語による評価では，従来のWAMによるコンパイルドコードに対して約4倍の処理速度の向上が得られ，58MIPSのRISCワークステーション上では3.3 MLIPSの高い処理速度が得られることを示した。また，RISCは汎用プロセッサであるにも関わらず，1推論あたりのマシンサイクル数が15〜20程度と，PROLOG専用機が実行に要する以下の，少ないマシンサイクルで効率良く処理を行えることを示した。

第4章では，PROLOGインタプリタの最適化方式について述べた。最適化方式としては，(1)インデキシング，(2)変数操作の最適化，(3)選択点生成の抑制，(4)環境生成の抑制といったコンパイラに用いられている4種類の最適化方式をインタプリタの実現方式に取り込む方式を提案し，コンパイラと同様な処理速度とメモリ効率の向上をはかっている。また，メモリ効率と処理速度を向上させるメモリ操作方式として構造参照法を提案し，評価によりその有効性を示した。これらの最適化方式を採用したPROLOGインタプリタの評価では，従来方式のインタプリタの4倍の処理速度と大幅なメモリ効率の向上が得られ，インタプリタでありながら，22MIPSのワークステーション上で62 KLIPSの高い処理速度を実現できることを示した。

第5章では，動的CDRコーディング方式について述べた。この章では，WAMの実現方式の特徴を効果的に利用して，リストを配列状に圧縮表現することにより，メモリ効率を2倍に高める方式を提案した。提案方式は，CDRコーディングの可能性を動的に検査するために，CDRコーディングの成功率を大幅に向上できるという特長がある。また，従来のCDRコーディング方式は非常にオーバーヘッドが大きいという問題があったが，RISCプロセッサ

のパイプラインの遅延スロットを効果的に利用することによって，オーバーヘッドを８％程度に抑える方式を提案しており，汎用のプロセッサでも効率よく実装できることを示した。さらに，ＣＤＲコーディングによってワーキングセットが縮小されることにより，使用するデータ量が大きくなってもキャッシュのヒットミスが増加せず，従来の方式に比べて高い性能を維持できることを示した。

第６章では，ＰＲＯＬＯＧの宣言的な表現能力をさらに高める制約評価方式について述べた。変数に付加された制約を管理するために，制約情報セルと呼ぶデータ構造を用いる方式を提案し，評価によって制約操作のオーバーヘッドが４％程度に抑えられることを示した。制約評価方式を導入したＰＲＯＬＯＧでは，変数の具体化と条件検査の順序に制限がないために，ゴールの実行順序を気にせず，より宣言的にプログラムを記述できるようになるという利点がある。また，指定した制約条件の下で解を探索し，変数が具体化された時点ですぐに条件の検査を行う処理が可能になるために，生成検査法で生じる探索空間の爆発を防ぐことができ，12Queens による評価では処理速度を7,800倍ほど向上できることを示した。

第７章では，本研究で得られた主な成果をまとめ，今後の課題について検討した。

論文審査の結果の要旨

本論文は，ＰＲＯＬＯＧ処理系の処理速度，メモリ効率，表現能力を向上させることに主眼をおき，最近広く利用されているＲＩＳＣ（Reduced Instruction Set computer）への実装を前提として，(1)コンパイラの最適化，(2)インタプリタの最適化，(3)動的ＣＤＲコーディング，(4)言語の表現能力の拡張の４つの処理方式に対して行った研究内容をまとめたものである。本研究に基づく提案方式は，(1)メモリアクセス頻度を低減する，(2)分岐処理を低減する，(3)パイプラインの空きを有効に利用するなどの，ＲＩＳＣアーキテクチャの高速性を効果的に引き出す実現方式となっている。

ＰＲＯＬＯＧコンパイラの最適化方式に関する研究では，ＰＲＯＬＯＧ処理系の基本的な実現方式であるＷＡＭ（Warren Abstract Machine）の抽象命令セットの機能構成と，ＰＲＯＬＯＧの大きな特徴である単一化処理と後戻り処理に着目し，(1)抽象命令の特殊化と複合化による最適化，(2)出力モードの単一化処理の最適化，(3)非決定性処理の除去による最適化の各方式を提案し，評価によりその特性と効果を明らかにしている。提案方式の append／３述語による評価では，従来のＷＡＭによるコンパイルドコードに対して約４倍の処理速度の向上が得られ，58 MIPS のＲＩＳＣワークステーション上では3.3 MLIPS の高い処理速度が得られることを明らかにしている。また，実現対象であるＲＩＳＣは汎用プロセッサであるにも関わらず，１推論あたりのマシンサイクル数が15～20程度と，従来のＰＲＯＬＯＧ専用機が実行に要する以下の，少ないマシンサイクルで効率良く処理を行えることを明らかにしている。

ＰＲＯＬＯＧインタプリタの最適化方式に関する研究では，(1)インデキシング，(2)変数操作の最適化，(3)選択点生成の抑制，(4)環境生成の抑制といったコンパイラに用いられている４種類の最適化方式をインタプリタの実現方式に取り込む方式を提案することによって，コンパイラと同様な処理速度とメモリ効率の向上をはかっている。また，メモリ効率と処理速度を向上させる構造参照法と呼ぶメモリ操作方式を提案し，評価によりその有効性を明らかにしている。これらの最適化方式を採用したＰＲＯＬＯＧインタプリタの評価では，従来方式のインタプリタの４倍の処理速度と大幅なメモリ効率の向上が得られ，インタプリタでありながら，22MIPS のワークステーション上で62 KLIPS の高い処理速度を実現できることを明らかにしている。

動的ＣＤＲコーディング方式の研究では，ＰＲＯＬＯＧでリストが多用されることに着目し，リストを配列状に圧縮表現することにより，メモリ効率を２倍に高める方式を提案している。提案方式は，ＣＤＲコーディングの可能性を動的に検査するために，ＣＤＲコーディングの成功率を大幅に向上できるという特長がある。また，従来のＣＤＲコーディング方式は非常にオーバーヘッドが大きいという問題があったが，ＲＩＳＣプロセッサのパイプラインの遅延スロットを効果的に利用することによって，オーバーヘッドを８％程度に抑える方式を提案しており，汎用のプロセッサでも効率よく実装できることを明らかにしている。さらに，ＣＤＲコーディングによってワーキングセットが縮小されることにより，使用するデータ量が大きくなってもキャッシュのヒットミスが増加せず，従来の方式に比べ

て高い性能を維持でき，最高で40%ほど性能が高い点が得られることを明らかにしている。

言語の表現能力の拡張に関する研究では，ＰＲＯＬＯＧの宣言的なプログラミング機能をさらに高める制約評価方式に関する研究を行っている。変数に付加された制約を管理するために，制約情報セルと呼ぶデータ構造を用いる方式を提案し，評価によって制約操作のオーバーヘッドが４%程度に抑えられることを明らかにしている。制約評価方式を導入したＰＲＯＬＯＧでは，変数の具体化と条件の検査の順序に制限がないために，実行順序を気にせずに，より宣言的にプログラムを記述できるようになるという利点がある。また，指定した制約条件の下で解を探索し，変数が具体化された時点ですぐに条件の検査を行うような処理が可能になるために，生成検査法で生じるような探索空間の爆発を防ぎ，12Queens による評価ではプログラムの処理速度を7,800倍ほど向上できることを明らかにしている。

以上のように本論文は情報工学研究の基本的ツールであるＰＲＯＬＯＧ処理系の性能向上に対する具体的な指針を示しているので，情報工学，特に知識情報処理の発展に寄与する所が大きい。よって本論文は博士論文として価値あるものと認める。