

58 The Aurora Or-Parallel Prolog System

D.H.D.Warren(Argonne National Lab.,U.S.A.)

発表要旨

AuroraはGigalips Projectとして知られる共同研究の一部として、シェアドメモリ・マルチプロセッサのためのプロローグパラレル実装のプロトタイプである。現在はシークエントでエンコアなマシンの上で動いている。Auroraはさらに現存する中でも、使いやすい、洗練されたシークエンシャル・プロローグ・システムであるSicstusプロローグを採用して作られている。マルチプロセッサバージョンを作る技法としてパイオニアシステムであるANL-WAMを採用している。SRIモデルはオアプロローグのオペレーションのためにSicstusプロローグエンジンを拡張する目的で採用された。この論文では現在のAuroraシステムのデザインと実装について述べ、評価している。その結果高いプロセッサ当たり絶対速度が得られ、実際の例で大きな実行速度の改善がみられた。その評価により、以後のシステム開発の方向と将来の研究の方向を決定した。

59 Cut and Side-Effects in Or-Parallel Prolog

B.Hausman(SICS,スウェーデン)

発表要旨

マルチプロセッサシステムの上でプロローグを実装するには二つの対照的なアプローチがある。一方のアプローチは別のマシンへ割り振る方法でその言語は厳密にプロローグと書える。他方のアプローチは新しい実行戦略のためのより適した新しい補足的な論理を作る方法である。この論文ではいくつかの新しい補足的な論理述語を紹介し、それと対応するプロローグの述語と比較する。プロローグの入出力、データベース、カット述語のためのいくつかの実装案についても議論する。最後に、実装の複雑さと許容される論理的な仕事量との間の好ましい折衷案である、カットの実行のアルゴリズムについて提案する。結論としてはプログラムがシークエンシャルでもパラレルでも、普通の副作用を持つ述語とカットの実装では同じオーダでは同じ性能が保証された。またこのカットの実装法では深く密集した探索木を持つ大きなプログラムに適している。このアルゴリズムによって起こるコンパイル時のみが問題である。