

異機種分散環境上の並列仮説推論システム

1U-7

中村 智典† 加藤 昇平‡
†名古屋工業大学

世木 博久† 伊藤 英則†
‡豊田工業高等専門学校

1 はじめに

本稿では、コストに基づく仮説推論を、ワークステーションクラスタによる仮想並列計算機で並列実行する方法について検討する。仮説推論は、不完全な知識の下で適切な推論を行う高次推論の一形式であり、柔軟で知的な推論をおこなう。仮説推論の問題として、推論処理の計算量が極めて大きいことが指摘されており、これを並列化することにより高速に解く手法について多くの研究結果が報告されている [2]。しかし、従来のシステムには、特定の並列計算機上でしか動作しないという問題点がある。

一方で近年、ワークステーションの能力の向上と、LAN 技術の発達により、高速なワークステーションによるネットワークが広く普及し、ネットワークを用いた並列処理が目されるようになった。このような環境では、ネットワークを構成する個々のワークステーションの処理能力が一定ではないことや、処理能力が動的に変化するため適切に負荷分散を行うことが困難である。

我々は、様々な処理能力のワークステーションから構成されるネットワーク上で効率よく動作する並列仮説推論システムを提案している [1]。本稿では、主にそのシステムが行う動的な負荷分散方法について述べる。また、実験により本負荷分散方法の有効性について報告する。

2 コストに基づく仮説推論

本稿では、仮説の選択の基準として知識ベースに含まれる各仮説に重み（コスト）が与えられている場合を対象に、与えられた観測に対して最適な説明を求め仮説推論を考える [3]。

知識ベースとして事実と仮説の知識、および観測が与えられる。事実だけでは観測が証明できないときに、真偽がはっきりしない仮説を真と仮定することにより観測を証明する。このとき用いられた仮説の集合を説明と呼び、この説明のコストが最小となるような組合せを求める推論である。推論を効率よく行うために、推論過程においてゴール節の評価関数を考え最適化を計っている。詳細については文献 [3] を参照されたい。

3 並列仮説推論システム

3.1 システムの概要

本節では、異機種分散環境上で動作する並列仮説推論システムのアルゴリズムについて説明する。仮説推

論の探索空間を複数のワークステーションに分散し、各ワークステーションが同期、通信を繰り返し並列に推論することにより最適な説明を求めるものである。仮想並列計算機を構成する個々のワークステーションは、通常マルチユーザー、マルチタスクシステムであるため、仮説推論の処理中にそれとは関係のないプロセスが動作することがある。そのため、推論処理のために仮想並列計算機中を構成する個々のワークステーションが持つ計算機リソースの全てをいつも利用可能とは限らない。そこで、本システムでは、動作中のワークステーションの状態を推定し、システム内のワークステーションの状態が均一になるように負荷分散を行う。

図1に本システムの構成を示す。システムは、マスターとスレーブの2種類のワークステーションからなる。スレーブは、ゴール節の展開、無矛盾性の検査、送信、受信を繰り返しながら与えられた観測を根とするSLD反駁木を生成する。マスターは、スレーブから送られてくる負荷情報からスレーブの状態を推定し、その状態に応じた量の探索空間をスレーブに割り当てる。

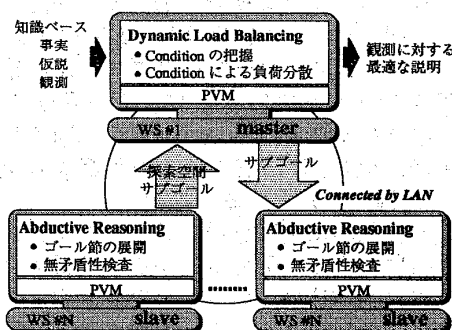


図1: システムの構成

3.2 動的負荷分散

異機種分散環境上で効率よく推論を行うためには、動的に処理能力が変化するスレーブに対し、その能力に応じて適切に負荷を分散する必要がある。また、並列ヒューリスティック探索においては、評価値のよいゴール節から順に探索を行うので、特定のスレーブに評価値の良いゴール節が集中することは避けたいと考えられる。そこで、本システムではスレーブの状態を以下の2点から計算する。

スレーブの処理能力 (*ability*) あるスレーブがどのくらい推論を行う能力があるかを表す数値。

ゴール節の輻輳 (*convergence*) あるスレーブに評価値のよいゴール節がどのくらい集中しているかを表す数値。

ある時点でのスレーブの処理能力は、最後に負荷分散を行ってからその時点までの間 (ユーザが指定した

A Parallel Cost-based Abductive Reasoning System on Heterogeneous Distributed Environments
Tomonori Nakamura, Shohei Kato, Hirohisa Seki and Hide-nori Itoh
†Nagoya Institute of Technology,
‡Toyota National College of Technology

パラメータ *INTERVAL* による一定時間) にスレーブが生成した部分 SLD 反駁木のノード数を基に推定される。

定義 3.1 i 回目の反復における, スレーブ j の処理能力の推定値を以下の式で定義する。

$$ability_i^j = \frac{|T_i^j|}{\sum_{s=1}^N |T_i^s|}$$

ここで, N はスレーブの台数, T_i^j は i 回目の反復でスレーブ j が生成した部分 SLD 反駁木, $|T_i^j|$ は T_i^j の大きさを表す。□

ゴール節の輻輳は, スレーブ中にあるゴール節の評価値の平均から推定される。

定義 3.2 CS_i^j (convergence set) を i 回目の反復時にマスターがスレーブ j から受信したサブゴールの集合のうち, 最良 $\alpha \times ability_i^j$ 個の要素集合とする。 α はユーザが与える定数である。ただし, スレーブ j が送信したサブゴールの数がこれに満たないときには, そのサブゴール全てを SC_i^j とする。この時, i 回目の反復時にスレーブ j にどのくらい評価値のよいサブゴールが集中しているかの度合いを次のように定義する。

$$convergence_i^j = \frac{\sum_{g \in CS_i^j} g \text{ の評価値}}{|CS_i^j|}$$

ここで, $|CS_i^j|$ は, CS_i^j の要素の数である。□

定義 3.3 i 回目の反復における, スレーブ j の状態の推定値は以下の式で定義される。

$$condition_i^j = ability_i^j \times convergence_i^j \quad \square$$

マスターはスレーブの *condition* を用い図 2 に示すアルゴリズムにより負荷を分散する。同図において *Slaves* はスレーブの集合を表し, *SG* はスレーブからマスターへ送信された全てのサブゴールの集合を表す。同アルゴリズムでは *SG* に含まれる全てのゴール節が分散されることは保証されない。同アルゴリズムの適用後, *SG* 中に存在するゴール節はスレーブの *ability* に比例して分散する。

このように動的に負荷分散を行うことで, スレーブの処理能力に見合った量のゴール節を分散でき, 特定のスレーブに評価値のよいゴール節が集中することを防ぐことができる。

4 実験結果

本稿で提案した動的負荷分散の有効性を調べるため, 動的負荷分散を用いたシステム (A) と, 静的な負荷分散¹を行うシステム (B) の性能の比較実験を行った。例題は n -bit 全加算器故障診断問題を用いた。回路を構成する論理素子は, 「正常」「開放」「短絡」の 3 つの状態をとるものとし, これを仮説として表現した。A, B とも同じ機種ワークステーション (SPARCclassc 32M) を接続したネットワーク上で動作させ, このうちの 1 台を過負荷状態にした。システムのパラメータは *INTERVAL*=2.3 秒, $\alpha = 10$ とした。表 1 に 20-bit 全加算器故障診断問題の結果を示す。同表においてス

¹本システムにおいて *ability* と *convergence* の値を定数にすることにより実現できる。

²例えば 16' は, 1 台のマスター, 15 台の普通のスレーブ, 1 台の過負荷状態のスレーブからなることを表す。

```

1 begin
2   while(Slaves ≠ ∅ and SG ≠ ∅)
3     begin
4       SG の中で最も評価値の小さいゴール
         を Gmin とする;
5       j := {∀s ∈ Slaves | ∀s' ∈ Slaves:
              conditionsj ≥ conditions'j};
6       Gmin をスレーブ j へ分散する;
7       SG := SG \ Gmin;
8       condition'j := conditionsj;
9       CSsj ∪ {Gmin} のうち最良 |CSj| 個の
         ゴール節を新しい CSj にする。
10      convergencesj を計算する
11      condition'j := abilitysj × convergencesj
12      if (condition'j = conditionsj)
         Slaves := Slaves \ j;
13    end
14  end

```

図 2: スレーブの *condition* による負荷分散

レーブ台数の右肩に ' がついているものはスレーブのうち 1 台が過負荷状態にあることを表す²。システム全体の能力を, 各スレーブの処理能力の和に等しい考えると, 推論時間は不等式

推論時間 $(N-1) < \text{推論時間}(N') < \text{推論時間}(N)$ を満たすべきである。しかし, B ではこの不等式を満たさず, 過負荷のスレーブを加えることでシステム全体のパフォーマンスを落としていることがわかる。

同表の「節展開数」は解発見までにシステムが生成したゴール節の数を表し, 探索空間の大きさに相当する。この値から, B は特に過負荷状態のスレーブを含む場合に A よりも多くの探索空間を探索し, その結果推論時間が多くかかっていることがわかる。

これらのことから, 我々が提案した動的負荷分散手法を用いることで過負荷状態のスレーブが混在している場合においても効率よく推論を行えることがわかる。

システム	A			B		
スレーブ台数	15	16'	16	15	16'	16
推論時間(秒)	158.1	143.6	137.6	175.6	183.6	161.6
節展開数	26201	25490	25335	29739	30622	26920

表 1: 実験結果

5 おわりに

本稿では, ネットワークに接続されたワークステーション上で, 効率よく並列仮説推論を行うための動的な負荷分散手法を提案した。本手法を適用することにより, システム中に過負荷状態のワークステーションがある場合においても, 効率よく推論を行うことができることを確認した。

参考文献

- [1] T. Nakamura, et al. A Parallel Cost-based Abductive Reasoning System on Heterogeneous Distributed Environments, *Proc. of 8th International Parallel Computing Workshop*, 1998. (to appear)
- [2] S. Kato, et al. Parallel Cost-based Abductive Reasoning for Distributed Memory Systems, *PRICAI'96, LNAI-1114*, pp.300-311, 1996.
- [3] 加藤, 世木, 伊藤, コストに基づく仮説推論における最適解探索の一手法, *情報処理学会論文誌 Vol.36, No.10*, pp.2380-2390, 1995