

ノード間交渉に基づく負荷分散方式の提案と評価

6B-2 小川 智之 小林 真也 木村 春彦 武部 幹
金沢大学 工学部

1. はじめに

ネットワークを介して複数の計算機を利用できるマルチコンピュータ環境において、計算機資源を有効に利用するための負荷分散方式が必要である。小規模なシステムに対しては、契約ネットプロトコル[1]等の分散問題解決のためのモデルが提案されている。しかし、現在のネットワーク・マルチコンピュータ環境においては放送型通信を頻繁に行う事は不可能である。そこで、我々研究グループでは1対1通信に基づく自律的な負荷分散方式[2]の提案を行ってきた。本稿ではこの自律負荷分散方式をシミュレーションにより評価し、その有効性について検討する。

2. 自律負荷分散方式

我々の提案した自律負荷分散方式は、未処理タスクを自ノードで行うか他のノードに依頼するかの判断(依頼判断)、及び他ノードからのタスク依頼に対する諾否判断を、各ノードが局所的情報から自律的に判断し、負荷を分散させる方式である。負荷分散の対象となるタスクは互いに独立しており、所要処理量(演算速度×所要時間)が予め予測可能なものである。

システムはネットワークに接続された複数のノードからなり、各ノードはタスクを実行する実行部、依頼・諾否を判断する判断部並びに負荷分散を行う際に必要となる種々の情報を管理するデータベースからなる。データベースには、他ノードの演算速度、自ノード対他ノードの通信路特性、最近の交渉で得た負荷情報(プロセス数、残余処理時間)を含む既知ノード情報が含まれる。各ノードは、以下のアルゴリズムに従い負荷を分散する(図1)。

a) 依頼判断 タスクの投入がされたノードは、各ノードでのそのタスクの所要時間を推定する。推定は、他ノードについては既知ノード情報から、自ノードについては実行部で現在処理中のプロセスの残余処理時間から行う。その後、ノード名、所要時間からなる候補リストを作成する。候補リスト中で所要時間が最小であるノードから順次依頼を行い、候補が自ノードとなるか、または依頼が受諾される

まで依頼を繰り返す。

b) 諾否判断 他ノードよりタスクを依頼されたノードは、まず自ノードについてそのタスク所要時間を推定し、これが依頼元ノードの所要時間より短ければ依頼を受諾する。そうでない場合には、さらに他ノードについて所要時間を推定し、候補リストを作成する。但しすでにそのタスクの依頼が行われたノードは除外する。これは、依頼の連鎖が巡回することを回避するためである。候補中に依頼元ノードの所要時間より短い所要時間となるノードがあれば依頼を受諾する。受諾後は、受諾したタスクに関して依頼判断を行う。これ以外の場合には依頼を拒否する。

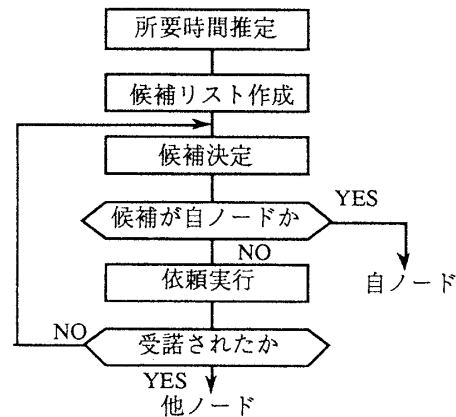
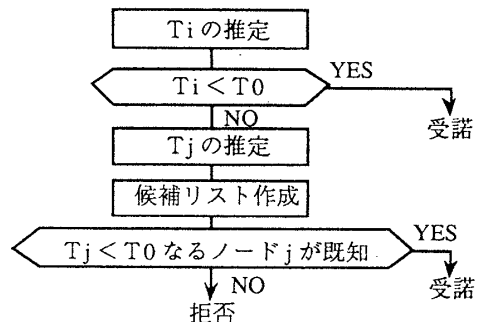


図1 a) 依頼判断



依頼元ノードの推定所要時間: T_0
自ノードの推定所要時間: T_i
他ノードの推定所要時間: T_j

図1 b) 諾否判断

Proposition and Evaluation of Task Allocation Method with Inter Process Negotiations

Tomoyuki Ogawa Shin-ya Kobayashi Haruhiko Kimura Tsuyoshi Takebe

Faculty of Technology, Kanazawa University

3. シミュレーションによる評価

シミュレーションにより自律負荷分散方式の評価を行う。まずシステム構成として、同一の処理速度（1ユニット/秒）であるノード10台（N1, N2, ..., N10）からなるシステムを想定する。またシステムに投入されるタスクは、平均1ユニットの指数分布に従うタスクサイズであり、平均到着率 λ でポアソン到着するものとする。また、通信遅延、依頼・諾否判断処理に伴う遅延は0としてシミュレーションを行った。

3.1 負荷分散状況

各ノードでのタスクの投入は、N1では到着率0.9, N2, ..., N10では到着率0.3とし、負荷分散を行った場合と行わなかった場合とについてのプロセス数（負荷）の時間変化を図2に示す。各ノードで負荷が2以上であった区間が負荷分散を行う事により大幅に1以下に抑えられ、アイドル状態であった区間が減少している事がわかる。例えばN1の利用率は0.822から0.637へ減少し、N2の利用率は0.369から0.627へ増加しており、提案方式によってシステム全体に渡り負荷が分散される事がわかる。

3.2 応答時間

N2, ..., N10のタスク投入の到着率0.3とし、N1の到着率 λ を変化させたときの平均応答時間を図3に示す。T1はN1の、T2はN2, ..., N10のタスク数5000個による平均応答時間である。負荷分散を行わない場合、平均応答時間T1は λ が1に近づくにつれ急激に増加するのに対し、負荷分散を行うと λ が5程度まででもT1は1.2程度に抑えられている。負荷分散を行った場合、T2もT1と殆ど同じ変化をしており、特に λ が5.5以上の場合、到着率の低い他のノードもこれに影響を受け、応答

時間が悪化している。しかしながら、これはシステム全体の処理能力に対して、システム全体に投入されるタスク量が過大なためである。

4. まとめ

本稿では、放送型通信を行わない自律負荷分散方式を均一なノード10台からなるモデルについてシミュレーションし、本方式により負荷が均一化され、応答時間が改善されることを示した。

ここでは、応答時間における実行時間以外の処理遅延を0としたが、依頼・諾否判断における処理遅延は、所要時間推定、候補リスト作成、依頼の連鎖による手間であり、各ノードの既知ノード情報数に依存し、実行するタスクのタスクサイズには依らない。従って、タスクサイズがこれらのオーバーヘッドに比べ大きい場合には、今回のシミュレーションと同様な結果が得られる。

また、均一なノードを仮定したが、本方式では既知ノード情報として演算速度も考慮に入れているため、不均一なノードからなるシステムにおいてもタスクを適切に分配できる。

今後、各ノードの処理能力が不均一な場合や、割り当てのためのオーバーヘッドが無視出来ない場合についても評価を行う予定である。

参考文献

- [1] Reid G. Smith: "The Contract Net Protocol: High-Level Communication and Control in a Distributed Problem Solver", IEEE Transaction on Computers, Vol.C-29, No.12 (1980)
- [2] 小川智之, 小林真也, 木村春彦, 武部 幹: "依頼による自律的な負荷分散方式の提案" 電子情報通信学会春期大会講演論文集 6-151 (1993)

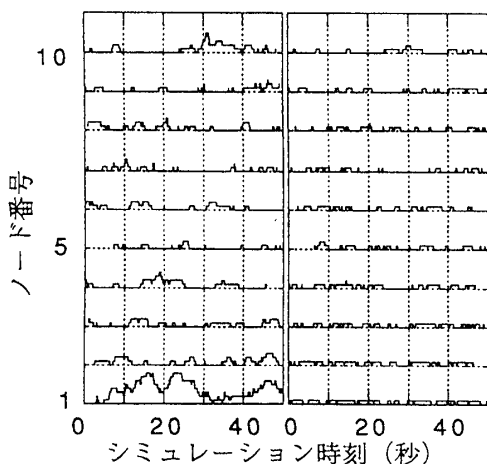


図2 プロセス数（負荷）の時間変化

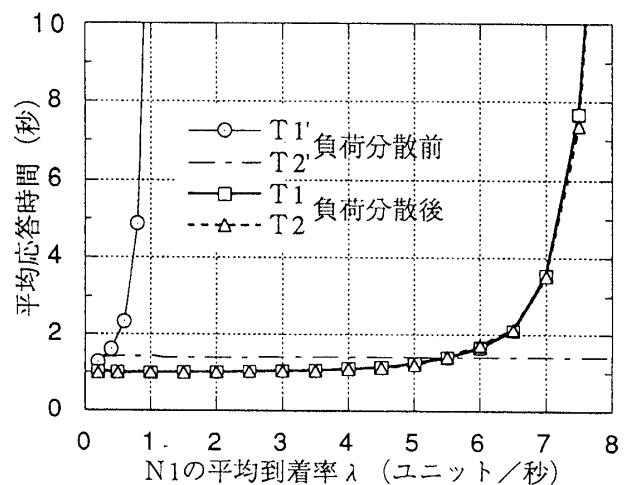


図3 応答時間—到着率