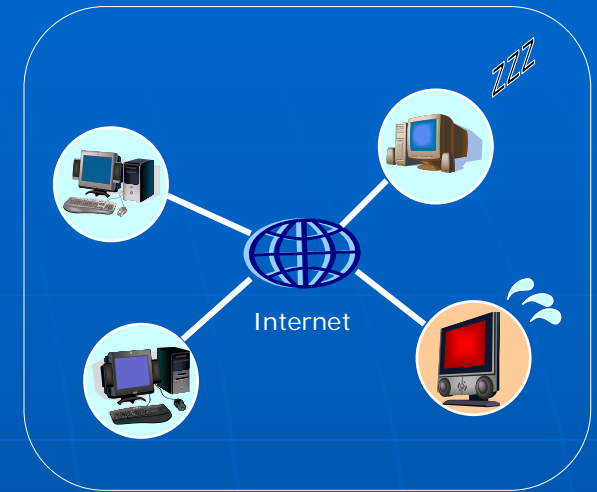


(グリッドスケジューリング・チームの研究概要)

時間変動する分散計算パワーを高效率で使うためのスケジューリング

研究の背景

- 使える計算パワーが時間変動するCPUを複数備えた並列分散計算環境の登場・普及
 - デスクトップ・グリッド, ボランティア・コンピューティング
 - インターネットに接続された家庭のPCは常にフル稼働しているわけではない
 - むしろCPUは遊んでいることの方が多い
 - ワープロや表計算, ネットサーフィンぐらいだとCPU利用率は低いまま
 - **この余っている計算パワー(余剰計算力)を集めれば莫大な計算パワーが得られる!**
- スーパーコンピュータの衰退とコモディティの台頭
- 莫大な計算パワーを要する重要な応用問題がたくさんある



Desktop Grid

元々のCPU性能がばらばらで, しかも計算速度が時間変動するCPUを大量(~数百万個)に備えた計算機を想像してください

研究の動機

グリッドの莫大な計算パワーは確かに魅力だが, 効率よく使うのはかなり難しいのではないかと?

- 各PCの余剰計算力は時刻の経過につれて変動する. しかもその予測は難しそう.
- 各PCのピーク性能(フルスピード)はバラバラ. PC以外の高性能な大型計算機も使えるとするとさらに差は広がる.

研究の目的

「われわれ人類は余剰計算力をどこまで使い切れるものなのか?」という深遠な問いに答えること

- 理論的な観点から
 - 「グリッド中の各PCに, いつ, どの仕事をやらせれば, 仕事全体が早く終わるか」というグリッド・スケジューリング問題の研究
 - グリッドのスケジューリング・モデルの提案
 - グリッド・スケジューリング問題の**近似アルゴリズム**の提案
 - グリッド・スケジューリング問題の**近似不可能性**に関する研究
- 実用的な観点から
 - 計算機の非専門家にも使いやすいグリッド・ミドルウェアの開発
 - 既存グリッド・ミドルウェアの**スケジューリング機構の改善の提案と実装**
 - **進化ロボティクス分野の実問題への応用**(ロボット研究者との共同研究)
 - **企業システムへのグリッド技術の応用**(関西電力との共同研究)

これまでの研究成果の一部

•理論的な成果

- 余剰計算力の変動に何らかの仮定を置かない限り、全体実行時間最小化グリッド・スケジューリング問題の近似アルゴリズムは存在しない(数学的証明) [1,2]
- グリッド全体での余剰計算力の消費量の最小化を目的とするグリッド・スケジューリング問題の提案 [1,2]
- 消費余剰計算力最小化グリッドスケジューリング問題の近似アルゴリズムの提案
 - 余剰計算力の時間変動の予測が全くできなくとも下記の性能を保証することを数学的に証明
 - 計算量が等しいN個の独立なタスクをM台のPCにスケジューリングする場合: $(1 + M \log M / N)$ -近似 [1,2] ($N \gg M$ ならほとんど最適)
 - 計算量が等しいN個のタスク(依存関係あり)をM台のPCにスケジューリングする場合: $(1 + cp(N) M \log M / N)$ -近似 [3] ($cp(N)$ はタスクグラフのクリティカルパス長で $cp(N) \leq N$ となる. $cp(N) \ll N$ かつ $N \gg M$ ならほとんど最適)
 - 余剰計算力の時間変動の予測がもしも完璧にできるなら
 - 計算量が様々なN個の独立なタスクをM台のPCにスケジューリングする場合: 2-近似 [4] (遅すぎるPCを自動検出してタスクを割り当てないというおもしろい性質を持つ)

•実践的な成果

- グリッド・ミドルウェアの開発
 - グリッドに計算パワーを提供する人、グリッドの計算パワーを利用する人の双方に使いやすい、非専門家向けグリッド・ミドルウェアを開発 [5]
- 「複数台で共同作業をするロボットの制御プログラムを進化的計算で自動獲得する」という実問題への応用
 - [1,2] のアルゴリズムをさらに発展させたものを実装し、大学間のPCを接続して評価実験を実施 [6]
 - 広島大学, 信州大学, 英バーミンガム大学との共同研究
- 企業利用向けグリッド・スケジューリングアルゴリズムの提案 [8] とメジャーな既存グリッド・ミドルウェア BOINC のスケジューリング機構の改善と企業アプリへの適用評価 [7] (関西電力との共同研究)

[1] N. Fujimoto and K. Hagihara. "Near-Optimal Dynamic Task Scheduling of Independent Coarse-Grained Tasks onto a Computational Grid". *International Conference on Parallel Processing (ICPP 2003)*, pp. 391-398, 2003.

[2] 藤本典幸, 萩原兼一. "グリッド上でのパラメータ・スイープ計算を対象として消費余剰計算力の最小化をねらった動的タスクスケジューリングのための近似アルゴリズム". 情報処理学会論文誌: 数理モデル化と応用, Vol. 46, No. SIG 10(TOM 12), pp. 1-9, 2005.

[3] N. Fujimoto and K. Hagihara. "Near-Optimal Dynamic Task Scheduling of Precedence Constrained Coarse-Grained Tasks onto a Computational Grid". *International Symposium on Parallel and Distributed Computing (ISPDC 2003)*, pp. 80-87, 2003.

[4] N. Fujimoto and K. Hagihara. "A 2-Approximation Algorithm for Scheduling Independent Tasks onto a Uniform Parallel Machine and its Extension to a Computational Grid". *IEEE International Conference on Cluster Computing (CLUSTER 2006)*, CD-ROM 7 pages, 2006.

[5] T. Noda, H. Mine, N. Fujimoto, and K. Hagihara. "A Parallel Computing Framework Not for Computer Experts: Easy Installation, Programming and Execution of Master-Worker Applications using Spare Computing Power of PCs". *International Symposium on Frontiers of Computational Science (FCS 2005)*, Springer-Verlag, to appear, 2007.

[6] 松浦芳樹, 大倉和博, 松村嘉之, 藤本典幸, 萩原兼一. "R3QIによる進化的計算の中粒度Gridスケジューリング". 情報処理学会論文誌: コンピューティングシステム, Vol. 47, No. SIG 12(ACS 15), pp. 240-249, 2006.

[7] 薄田昌広, 朝野郁郎, 藤本典幸, 萩原兼一. "企業内イントラグリッドにおけるジョブ管理機構の提案および検証". 第5回先進的計算基盤システムシンポジウム論文集(SACSIS 2007), ポスター発表予定, 2007.

[8] 峰久也, 藤本典幸, 東竜一, 薄田昌広, 萩原兼一. "余剰計算力を用いるグリッドのジョブスケジューリングにおける優先度制御の一手法". 情報処理学会研究報告, 2006-MPS-61, pp. 29-32, 2006.