指導教員	田浦健次朗 准教授	研究場所	工学部 2 号館

Big Computation, Big Dataの世界を切り開こう!

http://www.eidos.ic.i.u-tokyo.ac.jp/

1 はじめに

現在, 自然科学や工学のありとあらゆる分野で, コンピュータシミュレーションが必須の道具となっています (生体分子の働きの理解, 気候変動の予測, 新材料の設計, 銀河形成の過程の理解, など). それらは計算機の能力に飽くなき要求をし続け, いつの時代も大規模な並列計算によって支えられてきました. 身近な例では, 2011 年クイズ番組でチャンピォンを破ったコンピュータ (Watson) は, 約 3000 台の計算機が解の候補を生成し, 評価していました. また, Google のような巨大な検索エンジンの裏側では大量のデータを収集しそこから検索のための索引を作成する, ページの評価値を計算するなどの処理が, やはり並列計算によって行われています. 現在 (2012 年) 最高クラスの計算機は数千から数万の計算ノードを結合しています.

2 研究テーマ

我々の追求したい研究テーマの中心は, 高性能な並列計算を誰もが行えるようにするための, ソフトウェアです. 目指すは並列処理の「高性能」と「高水準」の両立です. 例えば,

- 超並列計算機のための新しい高水準並列プログラミング言語処理系
- 大規模データを高速に処理する新しい並列ファイルシステム・データベース
- プログラミングなしで並列処理を簡単に行うシェル, スクリプティング環境

などを設計, 実装し, 世の中で広く使われるソフトウェアを発信することを目指しています.

現在, CPU のクロック向上は頭打ちになっており, 逐次プログラムが CPU の性能向上で「(ソフトを書き換えなくても) 独りでに」速くなることはなくなりました. 今後は年々, 多くのコアを搭載した CPU が中心となり, それを有効活用できたソフトウェアのみが高速化されます. その意味でも今後のシステムソフトウェア研究は必然的に, 並列処理, 並列プログラミングのためのシステムソフトウェア研究でもあります.

3 進行中の研究プロジェクト (一部)

MassiveThreads 超軽量スレッドライブラリ: 通常のスレッドよりも二桁高速にスレッドを作れるスレッドライブラリです. それを土台にして, マルチコア計算機から, それらを多数結合した超並列機までを, 高水準にプログラミングできる言語を作る計画です (http://code.google.com/p/massivethreads/).

ParaLite 並列データベース: SQL が持つ高水準なデータ処理と, 既存のプログラムを容易に統合でき, かつそれを並列に実行できる並列データベースです. 大規模データ処理のための基盤システムです.

GXP 並列シェル・スクリプティング環境: 既存のプログラムを組み合わせた並列処理を, クラスタ, クラウドなど, 所構わずどこでも行えるようにした, 並列処理ツールです (http://www.logos.t.u-tokyo.ac.jp/gxp/).

研究成果は論文として発表するだけでなく, 実用レベルのソフトウェアとして構築し, オープンソースソフトウェアとして公開することを学生にも推奨しています.

4 計算機環境

- InTrigger (www.intrigger.jp) という, 17 の拠点を結んだ, 計 1800 CPU コアから成る環境を構築・運用しており, 大規模な分散環境での研究が可能です. 当研究室が構想から運用まで, 中心的な役割を果たしています.
- 東工大や東大の大型計算機を研究で利用することができ, 最先端スパコン上での研究が可能です.
- その他研究室内に 24/32 コアのマルチプロセッサなど,

並列処理の力, 楽しさを味わうのに最適な計算機環境があります.

5 どのような人が育ってほしいか?

まず多くの人がプログラミングができなくてはいけないと思っているかもしれません. 現実としてはある程度本当なのかも知れませんが, やや精神論を混じえて言わせてもらうなら, プログラミングの上手下手は決して話の中心ではない. どちらかというと, プログラミングがうまいからといって奢らず, 「自分の知らな分野をどんどん自分で勉強・吸収して行ける人」「常に自分で考える人」になってほしいと思っています. 並列処理の応用は広大で, それをいくらかでも深く知り, 設計や実装を進めなくてはいけません. 決して短時間で叶うことではありませんが, 幅広い分野を自分で学んでいける姿勢が重要です.