

田浦研究室 Taura Laboratory

Big Computation, Big Dataの世界を切り開こう！

◆田浦研HP → <http://www.eidos.ic.i.u-tokyo.ac.jp> (新4年生向け紹介文は トップ -> Research -> 新4年生向け紹介文)

田浦研とは？

研究室のメインテーマ

田浦研究室の中心的分野は、『**高性能な計算環境を容易に**(プログラマとして,あるいはエンドユーザーとして)利用でき,なおかつ**高性能**に実行できるような**基盤ソフトウェア**』です。そのために,「**プログラミング言語の設計**」,「**処理系・コンパイラ**」,「**実行時システム**」,「**オペレーティングシステムレベルの基盤技術の設計や実装**」,「**ファイルシステムやデータベースなどのデータ格納・処理基盤の設計や実装**」,「**プログラミングなしで並列処理を行うエンドユーザ向けツールの設計や実装**」,「**新しい並列アルゴリズムの研究**」などを行なっています。

B3のみなさんへ

多くのテーマは「**高性能な計算環境**」「**簡単に**」使えるようにするという点で共通しています。このゴールに興味・共感を覚え,自分でそこを変えようという意欲のある人を歓迎します。もちろん宗教ではありませんので,何に共感するかは人それぞれです。このゴールに共感を覚えなくても,**基盤ソフトウェア(言語やOS)**に興味がある人にはアドバイスが可能です。新しい並列アルゴリズムを作る,並列処理系を実応用に適用することに興味がある人も歓迎です。

研究内容

タスク並列処理系

■MassiveThreads

- 超軽量なスレッド(タスク)を膨大に生成し動的にコアに割り当てる,タスク並列モデルのプラットフォーム
- まさに「**高性能な計算環境**」を「**簡単**」に扱う基盤ソフトウェア

■MassiveThreads/DM

- 分散メモリ環境でタスク並列プログラムを実行できるプラットフォーム
※分散メモリとは?スパコンで用いられる形態で,多数の計算機ノードが高速なネットワークで接続されている
- 計算機ノードをまたがるポインタや,データの動的再配置・キャッシュをサポートし,プログラミングが容易に

■タスク並列システム向けベンチマーク

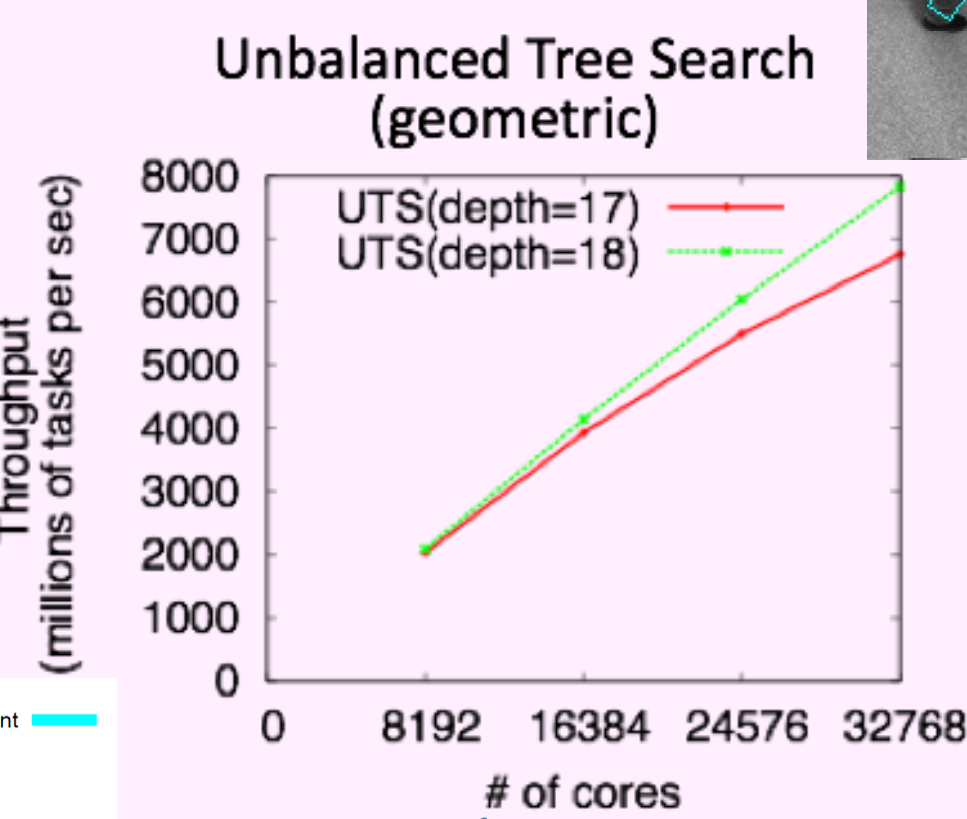
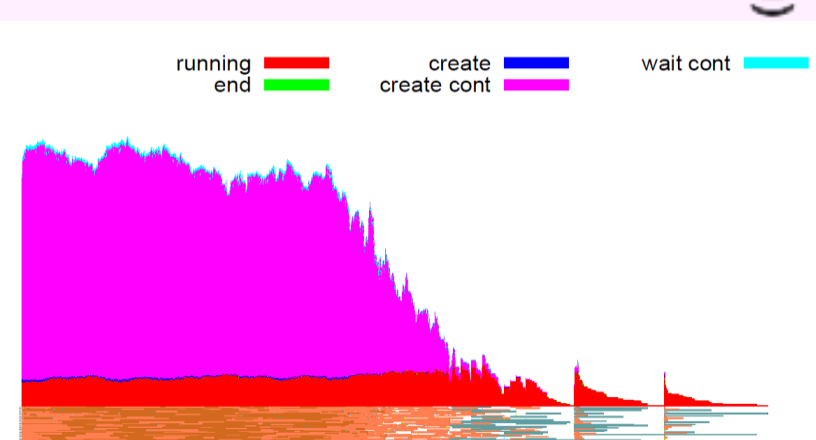
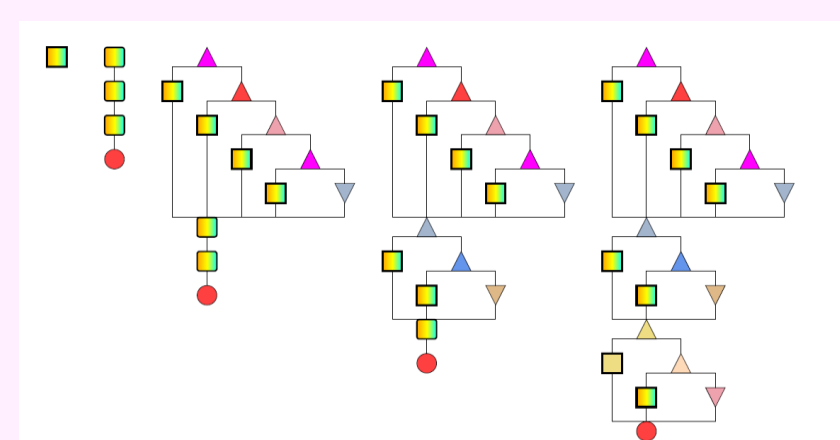
- 画像認識や機械学習といった,現実のアプリケーションに近いベンチマークであるPARSECを,様々なタスク並列処理系で実行できるラッパーを提供
- 発展を続ける様々なタスク並列システムを“実”性能・生産性の観点から比較・評価が可能に

■タスク並列向け性能モデリング

- DAGを用いた小規模でのプロファイリング情報を元に予測
- 実際には測定が困難なサイズのプログラムの評価を可能に

■タスク並列向けプロファイラ

- タスク並列プログラムを可視化・解析
- 性能低下の原因究明に有用
- 分散メモリ環境で動作するプロファイラ
- タスク並列プログラムの性能予測



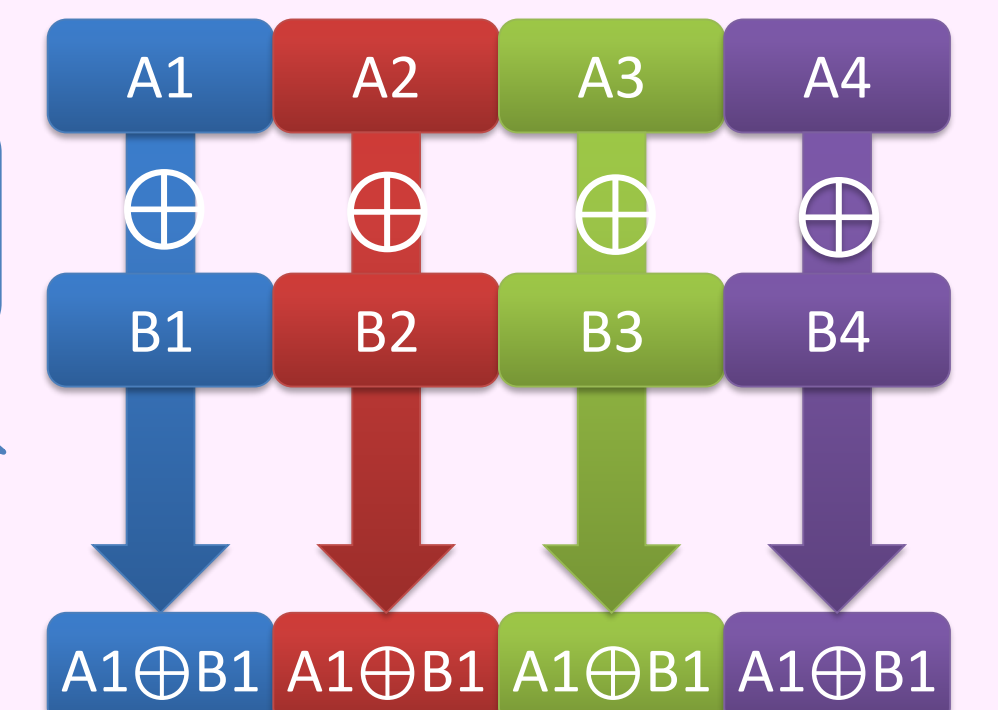
4400億個の軽量スレッドを生成, 3万プロセッサで良好な台数効果

並列アルゴリズム

※SIMD命令とは？

- 1命令で複数のデータを同時に処理

1命令で同時に4個の演算が可能



■SIMD命令を活用したアルゴリズム

- Burrows-Wheeler Transformアルゴリズムを改良, SIMDで扱いやすいようデータ構造を変えて高速化

■SIMD命令を活用するための言語処理系

- 静的タスクカットオフを利用した再帰関数のSIMD化
- Java仮想マシンのSIMD命令向け拡張

ファイルシステム

■次世代不揮発性メモリ向けファイルシステム AMFS

- キャッシュを介さずデータを読み込むことができ, アトミックな更新を用いて安全かつ高速にデータを永続化
- 電源喪失等によるクラッシュ状態をシミュレーションし, アプリケーションの安全性も評価

■広域分散環境上の分散ファイルシステム

- デスクトップPC・クラウド・クラスタなどの異なる環境間でも動作

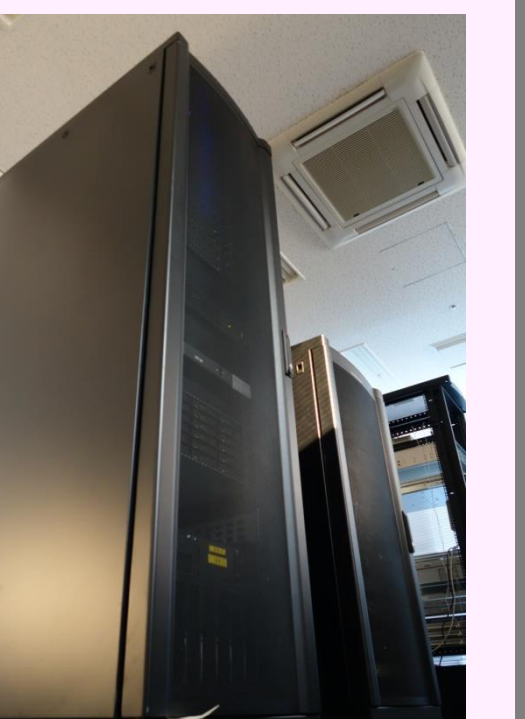
実験環境

■高性能サーバー

- 多数のコア(~72コア)や広大なメモリ(629GB)を持ったサーバー
- 他にもXeon Phi (60コア/240スレッド)×2なども

■分散環境

- 東大の保有するスパコン(FX10)での数万コア規模の実験



卒論生の生活

■特徴

- 先生がとても教育熱心
- 先輩方がとても優秀で優しい
- 成長するのにぴったりの環境です

■例えば、こんなことが身につきます

- 高性能なプログラムの書き方
- 基盤ソフトウェアへの深い理解
- Linuxマシンの扱い方

■ミーティング

- 週2回・基本的に午前中
- 進捗報告 (全体を2つに分けて交互に行う)
- Reading Circle(論文を1本読んで, スライドにまとめて発表)

■コアタイム

- ありません
- 自分のペースで研究できます

意欲のある学生皆様の参加を歓迎します^^わたしたちと一緒に充実した一年を過ごしましょう！

年間スケジュール表

4月	勉強会 (並列処理の基礎・高性能プログラミング)
5月	高性能プログラミング)
6月	テーマ決め
7月	院試
8月	
9月	中間発表
10月	
11月	研究
12月	
1月	
2月	卒論準備・発表



2016年度 卒論テーマ例

卒業論文のテーマは, 本人の興味のある分野に関連したものを相談しながら決定します

■FPGAの利用率向上を目的とした中間論理

■Locality-Aware Work Stealingの性能解析

■アトミックなデータ永続化を備えた次世代不揮発性メモリ向けファイルシステムの高速化

■大規模データ用の高速・省メモリ解析器の設計