High Performance Computing as a Combination of Machines and Methods and Programming

Soutenance en vue de l’obtention du Diplôme d’Habilitation à Diriger les Recherches

Claude Tadonki
Mines-ParisTech

Université Paris-Sud / 16 mai 2013
My Research Topics so far

- Fundamental Aspects of Algorithms and Complexity
  *(Formalism, proof, quantification, classification, ...)*
- Graph Theory and Applications (characterisation, modeling, scheduling)
- Discrete Dynamical Systems (Chip Firing Game, Periodicity, Garden of Eden, Invariants, ...)
- Parallel Scheduling (Theory, SPMD, SIMD, Systolic, ...)
- Polyhedral Model (Recurrences Equations and Scheduling)
- Efficient Parallel Programming (Hybrid Parallel Computing, Scalability, Load Balancing, Data Transfers & Exchanges)
- Automatic Code Generation and Transformations (High-Level Specification $\Rightarrow$ Program; Program + Annotations $\Rightarrow$ Program)
- Power Aware Computing (Energy Efficient Programming: Models and Methods, Cloud Computing)
- Applied Mathematics and Operation Research (Mathematical modeling, Optimization, Linear Algebra, Matrix Computation)

"A vouloir toujours aller au fond des choses, on court le risque d’y rester."
Places where I have been (study/work/conference/cooperation/teaching)

University of Yaoundé (Cameroon) → University of Rennes/IRISA → University of Geneva (HEC/CUI) → European Laboratory of Molecular Biology → University Paris-Sud → Ecole des Mines de Paris

92: CARI
97: CEPAMOQ

«science without border » is the hallmark of my route.

France  Italy  Spain  Argentina  Austria
Switzerland  Germany  Hungary  Madagascar  Brazil
Romania  Portugal  Finland  Senegal  UK
Danmark  Greece  USA  Morroco  Ireland
Canada  Mexico  Cuba  Egypt  India
Madagascar  China  Japan  Cameroon

Europe: CARI 92, CEPAMOQ 97.

France, Switzerland, Romania, Danmark, Canada, Madagascar, Italy, Germany, Portugal, Greece, Spain, Hungary, Finland, Russia.
Significant advances have been achieved in each of the aforementioned aspects. A skillful combination of all HPC components is really the key to **absolute efficiency**. This expected pluridiscipline interaction should be better done at the **earliest**. This is the **main point of my defense** and the motivation behind my **future plans**.
TRAVELING SALESMAN PROBLEM

Given a list of cities and their pairwise distances, the task is to find a **shortest tour** that visits each city exactly once.

\[
\begin{align*}
\text{minimize} & \quad \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij}x_{ij} \\
\text{subject to} & \quad \sum_{j=1}^{n} x_{ij} = 1 \quad i = 1, \ldots, n, i \neq j \quad (2.4) \\
& \quad \sum_{i=1}^{n} x_{ij} = 1 \quad j = 1, \ldots, n, i \neq j \\
& \quad x \in \{0,1\}^{n \times n} + \text{Subtour breaking constraints}
\end{align*}
\]

The TSP has an a priori \( n! \) complexity. Solving any instance with \( n = 25 \) using the current world fastest supercomputer (TITAN-CRAY XK7) might require 25 years of calculations. **TITAN in 1min \( \approx 6 \) billion people calculating**

"Tu quieres celeste que te cueste."

**Applications:**

Transportation, logistic, genome sequencing, benchmark for optimization methods, ...

TSP heros: Applegate, Bixby, Chvatal, and Cook

As difficult as the HAMILTONIAN CYCLE PROBLEM, which is **NP-COMPLETE** (Karp)

Please, forget about brute force approach! (16 cities \( \Rightarrow 653\,837\,184\,000 \) possibilities)

Modern optimization method have shown optimistic performances on practical instances

Competitive solutions are **parallel implementation of powerful optimization methods**

TSP (VLSI-Bell Labs) of size **85 900** solved in **1.5 year** (2004-2006) using a cluster of **96 2.8 GHz Intel Xeon** and **32 2.4 GHz AMD Opteron** connected with **100 MB ethernet**.

Approximation algorithm is also an interesting pragmatic way to go
Quicksort is a worst-case $n^2$ algorithm, but is still preferred to the $n \log n$ heapsort.

LP can be solved in a polynomial time, but the (exponential) SIMPLEX is still preferred.

The simplex has led to the leading solver CPLEX

We have implemented a nice interface between CPLEX and MATLAB in cooperation with David Musicant (Carleton College) and Travis Johnson.

Our interface (started in 2004) has been used and cited in several mathematical programming papers, including ours (P-median, Portfolio, Energy minimization).

Can be downloaded at
http://www.omegacomputer.com/staff/tadonki/using_cplex_with_matlab.htm
A Large-Scale Particles Physics Problem

- **High-precision** Lattice Quantum Chromodynamics simulations.
- The ANR project PetaQCD was targeting $256 \times 128^3$ lattices.
- One evaluation of the *Dirac operator* on a $256 \times 128^3$ lattice involves:

\[ 256 \times 128^3 \times 1500 \approx 10^{12} \text{ (stencil) floating-point operations} \]

### Curie Fat performance (weak scaling)

With 10,000 cores, we can roughly perform $500 \times 10^3 \times 10^6 = 5 \times 10^9$ fps.

Our $256 \times 128^3$ lattice would then require 200 seconds $\approx$ **3 minutes** for each evaluation of the *Dirac operator*.

Now, image that we have to do it 5000 times to solve One Dirac linear system !!!


---

**Claude Tadonki**
**Mines ParisTech**

High Performance Computing as Combination of Machines and Methods and Programming
HDR defense - University Paris-Sud/Orsay - PCRI - May 16th 2013
Important Facts about Supercomputers

The (peak/sustained) performance of supercomputers is increasing significantly ($\approx \times 10^5$ since 1993).

The following characteristics are becoming a standard:

- Several cores (2012 list: 84% $\geq 6$ cores and 46% $\geq 8$ cores)
- Vector units (with larger vector registers)
- Accelerated (mainly GPU: 62 systems of the 2012 including the #1)

The gap between peak and sustained performances on real-life applications is clearly questionable. HPC investigations should focus on this.

When it comes to accelerators, data transfer is critical. Compromise and overlap (scheduling)

TITAN CRAY-XK7 the (2012) world fastest supercomputer
- 299 008 CPU cores (16-cores AMD Opteron 6274)
- 18 688 NVIDIA Tesla K20 GPUs
- Peak: 27.11 PFlop/s.
- Sustained: 17.59 PFlop/s (Linpack)

Taking advantage of all aspects of a computing nodes requires a complex hybrid/heterogeneous programming. Code generation and transformation

Energy is major concern.
Power aware programming and scheduling.

Claude Tadonki
Mines ParisTech
High Performance Computing as Combination of Machines and Methods and Programming
HDR defense - University Paris-Sud/Orsay - PCRI - May 16th 2013
Let’s have a look on some of my achievements

"Only those who attempt the absurd...will achieve the impossible."  M. C. Escher
Given a graph \( G = (X, A) \) and a transition function \( \varphi \) such that \( \varphi(G) = (X, A') \) with \( \text{card}(A) = \text{card}(A') \). We study the evolution \( G_0 = G, \ G_1 = \varphi(G_0), \cdots, \ G_{n+1} = \varphi(G_n) \).

**Remark:** on a finite graph, such an evolution is ultimately **periodic**. Any state of the graph before the period is called **transient**. Interesting questions are

- the set of period lengths for a given graph
- bounds of period lengths
- garden of eden (graph with no transient length)
- relation between the structure of the graph and periodicity
- applications (games, simulation, graph scheduling)

This work gave an inspiration to derive a parallel scheduling methodology for acyclic graphs with some recurrent structure.

A system of recurrence equations (SRE) defining a variable $X$ has the following form:

$$X(z) = \left\{ D^N : g_i(...X(f(z))...) \right\}$$

$$F(i,j,k) = \begin{cases} \{i,j,k \mid k = 0\} : q_{i,j} \\
\{i,j,k \mid i = j = k\} : F(i,j,k-1) \\
\{i,j,k \mid i = k \neq j\} : F(k,k,k) \otimes F(i,j,k) \\
\{i,j,k \mid j = k \neq i\} : F(i,j,k-1) \otimes F(k,k,k) \\
\{i,j,k \mid i \neq k, j \neq k\} : F(i,j,k-1) \otimes (F(i,k,k) \otimes F(k,j,k-1)) \end{cases}$$

- Algebraic and syntactic transformations (polyhedral model framework, compiler, scheduler, symbolic analysis, ...)
- Graph based approaches easily apply to SRE through the underlying dependence graph ⇒ systematic synthesis!

Scheduling a system of recurrence equations onto $p$ processors is the task of finding a valid timing function $T$ (i.e. $T(z) > T(f(z))$) and an allocation function $A$ such that $[T(z_1) = T(z_2)] \Rightarrow [A(z_1) \neq A(z_2)]$.

**Theorem 1.** Let $G_1 = (X_1, \Gamma_1)$ and $G_1 = (X_2, \Gamma_2)$ two isomorphic graphs, and let $\varphi$ an isomorphism from $G_2$ to $G_1$. If $(t,a)$ is a valid schedule of $G_1$, where $t$ is a timing function and $a$ an allocation function, then $(t \circ \varphi, a \circ \varphi)$ is a valid schedule of $G_2$.

**Theorem 2.** Let $G = (X, A)$ be a directed acyclic graph. If $G$ is self isomorphic with the decomposition $X = X_1 \cup X_2 \cup \ldots \cup X_\gamma$ and the set of isomorphisms $\{\varphi_k, k = 2, \ldots, \gamma\}$, then the schedule specified by the timing function $t$ and the allocation function $a$ given by (6-7) is valid.

**Applications:** sorting, APP, Cholesky, tensor product of matrices

Our scheduling method (already published) opens the following perspectives:

- Implementation as a scheduling module into a polyhedral model based framework.
- Study of an appropriate graph clustering (hybrid parallelism or modularity)
- Inclusion of hardware parameters
Inside the LOGILAB, Operation Research Laboratory founded by Pr Jean-Philippe Vial and Pr Alain Haurie at the University of Geneva (Switzerland)

Dealing with modern, efficient, and somehow pragmatic approaches for modeling and solving difficult real-world problems

Think about gaussian pivoting vs iterative methods
Practical instances of discrete (pure or mixed) optimization problems are better solved though a skillfull combination of continuous optimization techniques and branch&bound-like mechanisms.

For a pure discrete problem, a relaxation is used. For a mixed formulation, a decomposition approach can be considered.

In number of cases, the objective function is (or becomes) non differentiable.

We then need a good non differentiable optimization method and solver.
ProxACCPM – Canonical Problem

\[
\min \{ f(y) = f_1(y) + b^T y + f_2(y) \mid y \in C \}, \tag{1}
\]

where

- \( f_1 \) and \( f_2 \) are convex functions on \( \mathbb{R}^n \) defined over \( C \subset \mathbb{R}^n \)
- \( f_1 \) is nonsmooth and \( f_2 \) is twice differentiable
- \( C \) is a convex domain
- \( b \in \mathbb{R}^n \) is a constant vector

**Remark 1** The nonsmooth function \( f_1 \) is often a positively weighted sum of \( p \) nonsmooth functions

\[
f_1(y) = \sum_{i=1}^{p} \pi_i f_{1i}(y). \tag{2}
\]

This property can be exploited in the method.

**Remark 2** For a given convex function \( f \), there are many combinations of the form \( f(y) = f_1(y) + b^T y + f_2(y) \). The appropriate choice is left to the user convenience.

Project funded by the **Swiss National Science Foundation**
In order to accomplish the optimization task, ProxAccpm runs in conjunction with user defined oracles.

An oracle is a user defined routine (black box for ProxAccpm), which computes and returns informations about feasibility and/or optimality.

We consider tree types of oracle:

**Feasibility oracle:** Check if the current point \( y \) belongs to the optimization domain. Otherwise, its returns a so-called feasibility cut, we denote \( (\xi, \tau) \in R^n \times R \), which satisfies

\[
\xi^T(y' - y) + \tau \leq 0, \forall y' \in C.
\]  

(3)

**First order oracle:** Given a feasible point \( y \), it returns \( f_1(y) \) and one subgradient \( u \in \partial f_1 \). This yields the following relation:

\[
u^T(y' - y) + (f_1(y') - f_1(y)) \leq 0, \forall y' \in C.
\]  

(4)

**Second order oracle:** Given a feasible point \( y \), it returns \( f_2(y), f'_2(y), \) and \( f''_2(y) \).
Localization Set: At a given step, the localization set is defined by

\[ \mathcal{L}_{\bar{\theta}} = \{(y, z, \zeta) \mid A^Ty - E^Tz \leq c, \pi^Tz + b^Ty + \zeta \leq \bar{\theta}, f_2(y) \leq \zeta, y \in Y_2\}, \quad (7) \]

where

- \( \bar{\theta} \) is the current upper bound
- \( A \) is the matrix of cuts (columnwise)
- \( E \) is a binary banded matrix

Methods differ in

- the management of the localization set (objective cut, updating, spatial transformations, etc...)
- the selection of the query point from the localization set
- the lower bound and the termination criterion
Accpm and Proximal Accpm

The Analytic Center Method: The query point is obtained by minimizing (over the localization set) the weighted logarithmic barrier given by

$$F(\vec{s}) = - \sum w_i \log s_i - \omega \log \sigma,$$

with $\vec{s} = (s_0, s, \sigma) > 0$ defined by

$$s_0 = \bar{\theta} - (\pi^T z + b^T y + \zeta),$$
$$s_i = c_i - (A^T y - E^T z)_i, \quad i \in I = \{1, \ldots, m\},$$
$$\sigma = \zeta - f_2(y).$$

The Proximal Analytic Center Method: The barrier function is augmented with a proximal term to yield the augmented barrier

$$\frac{1}{2} (y - \bar{y})^T Q (y - \bar{y}) + F(\vec{s})$$

where $Q$ is a positive definite matrix and $\bar{y}$ the so-called proximal center.
The main concerns are

- force the global convergence
- minimize the number of oracle calls
- optimize the computation time and memory space
- improve the computation accuracy
- take care of numerical stability

Achievements

- Implementation of the method in a complete framework (https://projects.coin-or.org/OBOE).
- Implementation as a query point generator for connection with other packages (branch and bound)
- Several case studies published in journal and conferences

Some perspectives

- Deep investigation of the parallelization on supercomputers (scheduling, scalability, load balance,...)
- Study how to deal with updates instead of performing matrix computations from scratch at each step
- Investigate on numerical issues with large-scale ill-conditioned systems
- Approximation algorithms
Energy Minimization

Power consumption is a crucial concern with embedded systems and supercomputers

Fujitsu K-Computer ≈ US$10 millions/year for electricity

My work on Power Aware Computing and Distributed Algorithms were initiated from my stay at the TCS-Lab.

- We focus on memory energy
- The memory is partitioned into several banks
- Each bank can be put into a specific power mode
- We assume regular (uniform) power state transitions
- We formulate and solve the optimization problem

\[
\begin{align*}
\text{min } & x^T Q y^T + R Q y^T \\
\text{subject to } & \\
1. & x \in \mathbb{N}^q, \\
2. & y \in \mathbb{R}^q, \\
3. & y_1 + y_2 + \cdots + y_q = p, \\
4. & y_0 \geq \mu C, \\
5. & x_1 + x_2 + \cdots + x_q = \rho C, \\
6. & y \leq \varphi x, \\
7. & y \geq \eta x.
\end{align*}
\]

Good collaboration with Mitali Singh and Viktor Prasanna (University of Southern California)
Energy Minimization

Input

Hardware parameters: \( W, Q, H, p, q, \text{ and } \delta \)
Program parameters: \( R, C \)
Memory management parameters: \( \rho, \varphi, \eta \)

Model and Optimization

Optimal Energy \( E \)
Optimal transition repartition \( X \)
Optimal time repartition \( Y \)

Output

Power aware program design and monitoring

Some perspectives are:

- Design a **methodology** that will use the output of our work for a systematic synthesis of **energy efficient policies**
- Extend and adapt our model to **current and future memory systems** (*multilevel* and *shared*)
- Use a similar formal approach to analyse the energy issue on the **cloud systems** (user and provider)
- Investigate on other approaches (dynamic scheduling, compilation, ...)

*One of our proposal on this topic received an important grant from the Swiss National Science Foundation*

We have been also investigating on:

- **Distributed algorithm in sensors networks** (*localization and information retrieval*)
- **Dual-power management problem** (*mathematical programming approach and heuristics*)
- **Algorithms for the web** (*search engines and social networks*)
PetaQCD – Overview

- ANR Project (HPC & Particles Physics)
- Origin of the universe (matter)
- Good cooperative effort
- Multidisciplinary collaboration
- HPC & numerical challenges


**Abroad:** K. Urbach, K. Jansen, L. Scorzatto, D. Pleiter, R. Tripiccione

*Lot of materials and outputs of this project can be found here*

https://www.petaqcd.org
PetaQCD – Contributions

- Hybrid extension of the reference package (tmLQCD) using Pthread library
- Accelerated algorithm & implementation of the Dirac operator and other linear algebra kernels

Illustrative results

- A 32×16³ configuration solved using the CGR algorithm on the PPU in 138 seconds
- The same configuration and algorithm on the (PPU + 8 SPEs) double precision in 4.58 seconds


- Block decomposition C. Tadonki, «Strengthening deflation implementation» for large scale LQCD inversions, Orsay, Sept. 27th – 28th 2012

We have implemented a generic block decomposition (multidimensional and with no restriction on the number of blocks per axis) within the tmLQCD package. The aim is to increase the basis of the deflation method (to solve large ill-conditioned systems) without

- Code generation

I start investigating on this topic from my collaboration with Dr Lionel Lacassagne (ANR project Ocelle 2007-2009), also SIMD and Image Processing stuffs.

Good collaboration with Dr Joel Falcou, Tarik Saidani, Khaled Hamidouche, and Pr Daniel Etiemble

DMA issues related to tiling

Performing the transfer expressed in figure 4 raises number of problems:

• **the region to be transferred is not contiguous on memory, thus list DMAs are considered**

• **the address of one given row is not aligned, thus the global list DMA is not possible**

• **the (address, volume) pair of a row does not match the basic DMA rules (the above two ones), thus the entire list DMA cannot be carried out**

• **misalignment could come from both sides (main memory and/or local store)**

• **the target region on the local store might be out of the container limits**

We have designed and implemented a routine which performs this task very efficiently
The Harris-Stephen algorithm is

- a corner (point of interest) detection algorithm
- an improved variant of the original algorithm by Moravec
- used in computer vision for feature extraction like
  - motion detection
  - image matching
  - tracking
  - 3D reconstruction
  - object recognition

Technically, the Harris algorithm is based on a pixelwise autocorrelation $S$ given by

$$ S(x, y) = \sum_{u,v} w(u) [I(x, y) - I(x - u, y - v)]^2 $$

where $(x, y)$ is the location of the pixel and $I(x, y)$ its intensity (grayscale mode).

<table>
<thead>
<tr>
<th>tile_h</th>
<th>tile_w</th>
<th>total time(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>512</td>
<td>0.0494</td>
</tr>
<tr>
<td>16</td>
<td>256</td>
<td>0.0698</td>
</tr>
<tr>
<td>32</td>
<td>128</td>
<td>0.0485</td>
</tr>
<tr>
<td>64</td>
<td>64</td>
<td>0.0345</td>
</tr>
<tr>
<td>128</td>
<td>32</td>
<td>0.0517</td>
</tr>
<tr>
<td>256</td>
<td>16</td>
<td>0.0699</td>
</tr>
<tr>
<td>512</td>
<td>8</td>
<td>0.0734</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>tile_h</th>
<th>tile_w</th>
<th>total time(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>512</td>
<td>0.198</td>
</tr>
<tr>
<td>16</td>
<td>256</td>
<td>0.238</td>
</tr>
<tr>
<td>32</td>
<td>128</td>
<td>0.187</td>
</tr>
<tr>
<td>64</td>
<td>64</td>
<td>0.110</td>
</tr>
<tr>
<td>128</td>
<td>32</td>
<td>0.180</td>
</tr>
<tr>
<td>256</td>
<td>16</td>
<td>0.218</td>
</tr>
<tr>
<td>512</td>
<td>8</td>
<td>0.352</td>
</tr>
</tbody>
</table>

Table 1. Timings on a 512×512 image

<table>
<thead>
<tr>
<th>tile_h</th>
<th>tile_w</th>
<th>total time(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>1200</td>
<td>0.494</td>
</tr>
<tr>
<td>10</td>
<td>600</td>
<td>0.360</td>
</tr>
<tr>
<td>20</td>
<td>300</td>
<td>0.264</td>
</tr>
<tr>
<td>40</td>
<td>150</td>
<td>0.235</td>
</tr>
<tr>
<td>80</td>
<td>75</td>
<td>0.183</td>
</tr>
<tr>
<td>160</td>
<td>37</td>
<td>0.247</td>
</tr>
<tr>
<td>320</td>
<td>18</td>
<td>0.275</td>
</tr>
</tbody>
</table>

Table 3. Timings on a 1200×1200 image

We observe **50% improvement** between square tiles and full row tiles.

We observe **50% improvement** between square tiles and full row tiles.

PetaQCD – Algebraic Path Problem

Warshall-Floyd + Shift-toroidal

Analytical expression of the shift-toroidal

\[ m_{i-1,j-1}^{(k)} = m_{ij} \oplus (m_{ik}^{(k-1)} \otimes m_{kj}^{(k-1)}) \]

Mapping of our algorithm on the Cell

- PPE-DMA is issued only by the first and the last processor
- Inner SPEs communicate and synchronize locally
- Computation-communication overlap occurs for all communications
- Can run on more SPEs or CELL Blades by natural extension

<table>
<thead>
<tr>
<th>Tile</th>
<th>1 SPE</th>
<th>2 SPEs</th>
<th>8 SPEs</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>t(s)</td>
<td>t(s)</td>
<td>t(s)</td>
</tr>
<tr>
<td>1</td>
<td>6.67</td>
<td>3.28</td>
<td>2.03</td>
</tr>
<tr>
<td>4</td>
<td>5.01</td>
<td>2.50</td>
<td>2.00</td>
</tr>
<tr>
<td>8</td>
<td>4.79</td>
<td>2.39</td>
<td>2.00</td>
</tr>
<tr>
<td>12</td>
<td>4.70</td>
<td>2.32</td>
<td>2.02</td>
</tr>
<tr>
<td>16</td>
<td>4.72</td>
<td>2.36</td>
<td>2.00</td>
</tr>
</tbody>
</table>

(c) Performance with a 1024x1024 matrix

1 Performance results

All our experimentations are performed with double precision data. We used a $32 \times 16^3$ lattice (hence 131,072 sites)

<table>
<thead>
<tr>
<th>#SPEs</th>
<th>time(s)</th>
<th>speedup</th>
<th>GFlops</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.109</td>
<td>1.00</td>
<td>0.95</td>
</tr>
<tr>
<td>2</td>
<td>0.054</td>
<td>2.00</td>
<td>1.92</td>
</tr>
<tr>
<td>3</td>
<td>0.036</td>
<td>3.00</td>
<td>2.89</td>
</tr>
<tr>
<td>4</td>
<td>0.027</td>
<td>3.99</td>
<td>3.85</td>
</tr>
<tr>
<td>5</td>
<td>0.022</td>
<td>4.98</td>
<td>4.73</td>
</tr>
<tr>
<td>6</td>
<td>0.018</td>
<td>5.96</td>
<td>5.78</td>
</tr>
<tr>
<td>7</td>
<td>0.015</td>
<td>6.93</td>
<td>6.94</td>
</tr>
<tr>
<td>8</td>
<td>0.013</td>
<td>7.88</td>
<td>8.01</td>
</tr>
</tbody>
</table>

Figure 1: $32 \times 16^3$ Wilson-Dirac on a QS20

<table>
<thead>
<tr>
<th>#SPEs</th>
<th>time(s)</th>
<th>speedup</th>
<th>GFlops</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.0374</td>
<td>1.00</td>
<td>2.76</td>
</tr>
<tr>
<td>2</td>
<td>0.0195</td>
<td>1.91</td>
<td>5.31</td>
</tr>
<tr>
<td>3</td>
<td>0.0134</td>
<td>2.79</td>
<td>7.76</td>
</tr>
<tr>
<td>4</td>
<td>0.0105</td>
<td>3.56</td>
<td>9.90</td>
</tr>
<tr>
<td>5</td>
<td>0.009</td>
<td>4.81</td>
<td>13.27</td>
</tr>
<tr>
<td>6</td>
<td>0.0081</td>
<td>5.75</td>
<td>15.87</td>
</tr>
<tr>
<td>7</td>
<td>0.0076</td>
<td>6.84</td>
<td>18.88</td>
</tr>
<tr>
<td>8</td>
<td>0.0075</td>
<td>7.82</td>
<td>21.59</td>
</tr>
</tbody>
</table>

Figure 2: $32 \times 16^3$ Wilson-Dirac on a QS22

<table>
<thead>
<tr>
<th></th>
<th>Without SSE</th>
<th>With SSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 core</td>
<td>0.0820</td>
<td>0.040</td>
</tr>
<tr>
<td>4 cores</td>
<td>0.0370</td>
<td>0.0280</td>
</tr>
</tbody>
</table>

Figure 3: $32 \times 16^3$ Wilson-Dirac timings (seconds) on an INTEL i7 quadcore 2.83 Ghz

SIMD implementation of the basic operators (intensive use of SPU intrinsics)

double buffering technique to overlap DMA and computation (ideal in double precision)
optimal list DMA organisation by an algorithm similar that used for the bin packing problem (significant latency reduction)

Cooperation with Brazil
- **BioCloud** project of the STIC-AmSud
- Sandwich PhDs (*Brazilia/Rio/Niteroi*)
- Papers, visitings, events, courses
- Other Latin America partners (*Chile, ...*)
- + INRIA and Paris-Sud University

Cooperation with Morocco (*Oujda*)
- Sandwich PhDs
- Papers, visitings, seminars

Cooperation at Mines ParisTech
- TIMC project (multi-target images)
- CMM (Math Morphology) and CAOR

**CPOCC**
1st Workshop on Cost and Performance Optimization in Cloud Computing
in conjunction with the
26th International Symposium on Computer Architecture and High Performance Computing
October 23-26, 2013, Porto de Galinhas, Pernambuco, BRAZIL

Cloud computing is coming in vogue as a convenient solution to share high performance computing resources (CPUs, storage media and applications) among a widespread set of users. This configuration brings number of advantages related to the cost from both end-users and providers sides. It also offers a great opportunity to have a better control over power consumption and carbon emission. The concept of cloud computing itself is not so recent, but its implementation considering the various kinds of available devices is still on the way to maturity. The increasing audience and quality of service issues related to cloud systems stress to investigate on cost and energy at various levels. From the user side, the pay-as-you-go model together with the possibility to choose the resource configuration offer a way to reduce both cost and wastage. From the provider point of view, energy remains the main concern, which needs to be optimized for both profit and lifetime. Therefore, we need good models and methodologies that help to manage the cloud efficiently according the aforementioned keypoints, with a special attention to cloud federation.

CPOCC workshop aims at providing a platform to present and discuss both theoretical and experimental investigations related to Cloud computing performance and cost analyses. Authors are invited to submit original manuscripts on any of the topics described below.

**TOPICS**
- Topics of interest for the workshop include but are not limited to:
  - power consumption models for clouds and analysis
  - cost models and optimization strategies for cloud computing
  - federated cloud models and benchmarking
  - successful case studies on existing clouds
  - data and compute intensive applications on the cloud (bioinformatics, ecology, weather and climate models, astrophysics, computational finance, computational chemistry, large databases, and others)
  - cloud federations

**WORKSHOP ORGANIZATION**
- Claude Tadonki (Mines ParisTech / France)
- Christine Eisenbers (INRIA / France)
- Luísa Maria Drummond (Federal Fluminense University / Brazil)
- Alba Cristina Magalhães Alves de Mota (University of Brasilia / Brazil)
- Maria Emília Machado Telles Walter (University of Brasilia / Brazil)
- Mario Inostroza-Ponta (University of Santiago de Chile / Chile)
- Carolina Bonacic (University of Santiago de Chile / Chile)

Location of the conference and workshops at Porto de Galinhas

Location of the conference and workshops at Porto de Galinhas

BioCloud Partners at Orsay
In memory of Jean Tadonki (1939-2001)
« You are the greatest »
Thanks to all of you

*Family, friends, colleagues, advisors, examiners, referees, administration, technicians, audience, you, ...*