European Union
European Social Fund
OPERATIONAL PROGRAMME EDUCATION AND LIFELONG LEARNING
investing in knowledge society
MINISTRY OF EDUCATION & RELIGIOUS AFFAIRS
MANAGING AUTHORITY
Co-financed by Greece and the European Union
NSRF 2007-2013
EUROPEAN SOCIAL FUND

Hybrid PDE solvers

Manolis Vavalis and George Sarailidis
Computer and Communication Engineering Department
University of Thessaly, Volos, Greece
University of Thessaly

Objective

Long Term

- Investigate the prospects of combining the effectiveness, the versatility and the flexibility of Monte Carlo methods with the the rigorousness and the robustness of the Finite Element (or Finite Difference, or spectral) methods into truly added value numerical PDE solvers.
- Identify the theoretical and practical obstacles, elucidate the characteristics and idiosyncrasies of the proposed methods and investigate software development and engineering emerging issues
- Convince researchers and practitioners that such hybrid methods can be effectively used at large everyday scale.

Short Term

- Design a generic computational framework and develop an associated effective prototype hybrid solver for multi-domain linear elliptic PDEs in 2 and 3 dimensions.

This is a preliminary study of an on-going effort on multi-domain, multi-physics simulation systems.

Hybrid PDE Method

The user specifies the interfaces of the subdomain(s) is interested in.

Stochastic prepossessing Monte Carlo-based walks on spheres provide approximations of the solution at selected points on the interfaces to decouple the original PDE problem into a set of independent PDE sub-problems.

Interpolation smoothing Interpolation procedures use the computed Monte Carlo approximations at selected points on the interface to provide accurate enough boundary conditions to local PDE sub-problems.

Deterministic solving Selected finite element solvers compute independently the local solution to each (or selected) resulting sub-problems.

Elements of the above method can be found at [1, 3]

The Generic Algorithm

Data: $i_1, i_2, \ldots, i_N$: the ids of the subdomains in which we wish to compute the solution.

Result: $\tilde{u}_\mu$, $\mu = i_1, \ldots, i_N$: computer approximations of the restrictions of the exact solution u in the subdomains $\mathcal{D}_\mu$, $\mu = i_1, \ldots, i_N$.

```
// PHASE I: Estimate solution on the interfaces
```

while $\mathcal{I}_{\mu,\nu} \subset \cup_{j=1}^{N} \partial \mathcal{D}_{i_j}$ **do**

Select control points $x_i \in \mathcal{I}_{\mu,\nu}$, $i = 1, 2, \ldots, M_{\mu,\nu}$;

Estimate the solution u at control points x_i using a Monte Carlo method;

Calculate the interpolant $u^I_{\mu,\nu}$ of $u\mu, \nu$ using the control points x_i;

end

```
// PHASE II: Estimate solution in the subdomains
```

for $j = 1, 2, \ldots, N$ **do**

Solve the PDE problem:;

$L_{i_j} u_{i_j}(x) = f_{i_j}(x) \quad x \in \mathcal{D}_{i_j}$;

$B_{i_j} u_{i_j}(x) = g_{i_j}(x) \quad x \in \partial\mathcal{D}_{i_j} \cap \partial\mathcal{D}$;

$L_{i_j} u_{i_j}(x) = h_{i_j}(x) \quad x \in \mathcal{D}_{i_j}$; // construct $h_{i_j}(x)$ from $u^I_{\mu,\nu}$s

end

Our Specific Implementation

- **PHASE I: Estimate solution on the interfaces**
  - Compute estimations of the solutions at selected points
    - Our C++ implementation of a walk-on-spheres method [2]
    - Beyond proof of concept. Focus on efficiency and effectiveness.
    - Can be reused as a detached module
  - Use computed values to obtain the interpolant of the solution on the interface lines
    - 2D problems: Burkardt's C++ splines library
    - 3D problems: SINTEF's C++ Multilevel B-splines library
- **PHASE II: Estimate solution in the subdomains matches**
  - deal.II library

Implementation Details

- Modular, extensible, full multi-threaded C++ code with simple usage interface and API
- The first step of the MC method is the longest and quasi-randomness is taking advantage of its "uniformity". Easily cuts the high frequency terms of the error.
- Additional attention is needed to avoid the correlation of the quasi-random sequence of and technical issues effort is devoted
- Further quasi-randomness is not necessary and has not be utilized.

Configuration (input) file

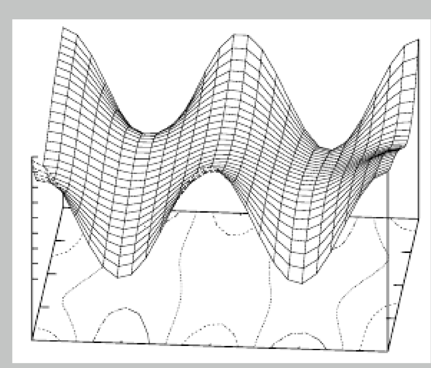
- General Parameters
  - 3 Number of dimensions (2 or 3)
  - 4 Maximum number of threads
- Geometry Parameters
  - 1 Dimension X length ...
  - u Uniform/non-uniform subdoms
  - 2 Subdomains along X ...
- PDE solver's parameters
  - 6 Grid refinement level ...
- Monte Carlo Parameters
  - 100 Number of walks
  - 10^{-6} Boundary tolerance
- Interpolation Parameters
  - 2 Interpolation nodes on the yz-interface ...
  - u Levels in the hierarchical construction of B-splines

Preliminary Results

- 2D Laplace, 4 uniform subdomains, error reduction in top-right

cycle	cells	dofs	L_2 norm	$L_\infty norm$
0	8	125	6.405e+00	2.102e+01
1	64	729	1.083e+00	4.249e+00
2	512	4913	1.366e-01	6.255e-01
3	4096	35937	1.848e-02	1.380e-01

- True solution



Synopsis & Prospects

- We have
  - Designed a software framework and implemented an associated prototype for hybrid PDE solvers.
  - Identified the importance of quasi-randomness for the first step (only).
  - Realized the necessity of properly balancing the accuracy of the random walks, the interpolant, the FE solver and the floating point arithmetic.
  - Convinced of the effectiveness of the meta-computing paradigm at the level of reusing state-of-the-art high performance numerical solvers (and beyond).
  - Confirmed the natural multi-threaded implementation and the associated natural mapping on modern many-core systems.
- We plan to
  - Continue experimentation and theoretically analyze model problems
  - Develop hybrid methods for the two engineering problems considered.
  - Extent implementation for general linear Elliptic PDEs on multi-rectangular domains
  - ...

References

J. Acebrón, M. Busico, P. Lanucara, and R. Spigler.
Domain decomposition solution of elliptic boundary-value problems via Monte Carlo and quasi-Monte Carlo methods.
SIAM Journal of Scientific Computing, 27(2):440–457, 2006.

J. M. DeLaurentis and L. A. Romero.
A Monte Carlo method for Poisson's equation.
J. Comput. Phys., 90(1):123–140, 1990.

M. Mascagni, A. Karaivanova, and Y. Li.
A quasi-Monte Carlo method for elliptic partial differential equations.
Monte Carlo Methods and Applications, 7:283–294, 2001.

Work partially support through the GSRT Thales Grand mmspmm http://mmspmm.inf.uth.gr/