Journal of Computational Physics 374 (2018) 91–120

Contents lists available at ScienceDirect

Journal of Computational Physics

www.elsevier.com/locate/jcp

Fast and scalable algorithms for constructing Solvent-Excluded Surfaces of large biomolecules

Raphael Egan [a,*], Frédéric Gibou [b]

[a] Mechanical Engineering Department, University of California, Santa Barbara, CA 93106, United States of America
[b] Mechanical Engineering Department & Computer Science Department, University of California, Santa Barbara, CA 93106, United States of America

ARTICLE INFO

Article history:
Received 24 January 2018
Received in revised form 16 July 2018
Accepted 17 July 2018
Available online 23 July 2018

Keywords:
Level-set
Adaptive refinement
Implicit solvent models
Solvent-Excluded Surface
Cavity identification
Parallel computing

ABSTRACT

We present algorithms for constructing a level-set representation of the Solvent-Excluded Surface of biomolecules. The algorithms are utilizing Octree Cartesian grids in the paradigm of distributed computing, and are designed to balance the computational load. The method is shown to be fast, scalable and first-order accurate. The procedure is robust, creates a feature-preserving surface and naturally handles any topological difficulty within its mathematical formulation. Several two- and three-dimensional illustrations are provided as well as accuracy and scaling analyses. We show the ability to construct computational grids for moderately large molecules in less than a few seconds when an optimal number of processors is used.

1. Introduction

Modeling solute-solvent interactions stands as a cornerstone within the plethora of algorithmic and numerical challenges associated with computational biomolecular physics: these interactions play a fundamental role in the estimation of free energies related to chemical processes, see e.g. [26,40]. Explicit solvent models, *i.e.*, treating solvent molecules explicitly and resolving their fundamental dynamics, are believed to give the most detailed description of such interactions. However, their computational cost quickly becomes intractable for large-scale systems. On the other hand, implicit solvent models treat the solvent as a continuum medium and capture the solvent-solute interactions through a classical electrostatic scalar field Ψ. Implicit solvent models have been used successfully and showed to be satisfactory in the last decades [28,35,57,63].

Mathematically, the electrostatic potential Ψ is a scalar field that obeys an exponentially nonlinear Poisson–Boltzmann Equation (PBE), in the context of implicit models allowing mobile ions in the solvent. The permittivity ϵ is discontinuous across the solvent-solute interface, with a discontinuity of at least one order of magnitude. Dirac-distributed source terms model (partial) atom electric charges. Finally, specific interface *jump conditions* at the solute-solvent interface impose the continuity of Ψ and $\epsilon \boldsymbol{n} \cdot \nabla \Psi$ across the interface, where $\boldsymbol{n}$ is the vector normal to the interface.

Starting from the late 1980s, several advances have been made to address this problem numerically, either using Finite Element/Volume (FE/FV) methods like APBS [1,33,34], Finite Difference (FD) methods [6,20,36,41,52], or even, more recently,

* Corresponding author.
E-mail address: raphaelegan@ucsb.edu (R. Egan).

https://doi.org/10.1016/j.jcp.2018.07.035
0021-9991/

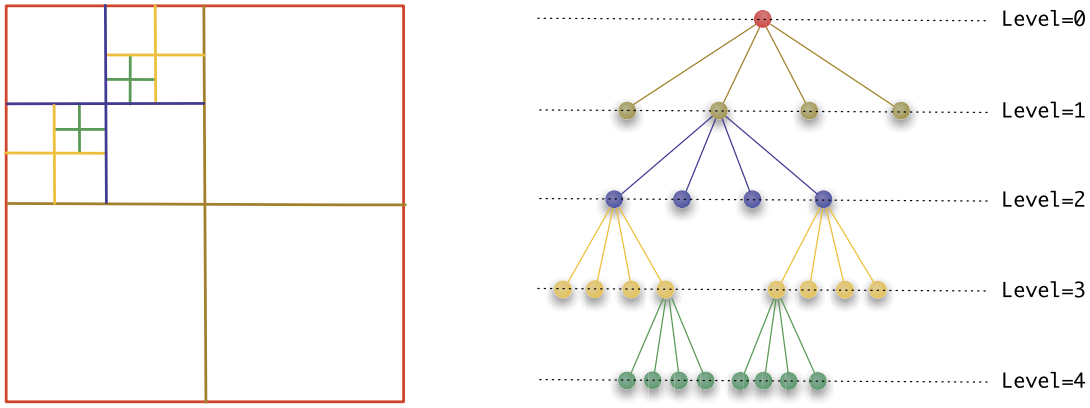

Fig. 1. Discretization of a two-dimensional domain (left) and its Quadtree representation (right). The entire domain corresponds to the root of the tree (level 0), and each cell subdivided further points to its four children.

Boundary Integral (BI) methods [9,10,71]. The linearized version of PB, referred to as Debye–Huckel theory, has been considered in some of these works; however, the model becomes accurate only when the solution ionic strength approaches zero, a contradiction with typical biochemical applications that involve relatively high ionic strengths. We refer the interested reader to the reviews [2,43] for more details about the PBE and related numerical aspects.

Because of their ability to enforce the continuity conditions at the interface and their natural capability for adaptive refinement, possibly with *a priori* error estimates, the FE/FV techniques soon outperformed the FD approach and thus became methods of choice. Indeed, original FD attempts were constrained to uniform grids; moreover, grid points were simply marked as *solvent-accessible* or not, and the same discretization stencil was applied throughout the computational domain, disregarding any conditions at the solute-solvent interface [42,56]. From a numerical point of view, such a strategy leads to poor performances because of undesirable grid-size-dependent numerical smearing of the interface, although sometimes supported *qualitatively* in the literature by overlap of wavefunctions in a quantum mechanics sense [16]. However, more recent developments in FD techniques have addressed the issues of the continuity conditions at the solute-solvent interface [14,31,72] and/or adaptivity [32,47,48]. Moreover, typical finite element approaches rely on body-fitted meshes that are not trivial to construct in general, especially when considering the constant remeshing needed to study folding of molecules. Finite differences, on the other hand, treat such cases straightforwardly so that there is nowadays a clear advantage to use this strategy. The level-set method introduced by Osher and Sethian [53] stands as a linchpin in this context: relevant interfaces are implicitly represented by the zero-level set of a function ϕ. We refer the interested reader to [27] for a recent review of the capabilities of level-set methods.

When adaptivity comes into play, the ability to construct an adaptive mesh efficiently in a distributed computing framework is a key feature for efficient and/or large-scale numerical simulations. Such a mesh may be built upon a triangulation of the solvent-solute interface, which can be constructed efficiently by optimized tools like the `MSMS` [59] (although unable to process very large molecules, see [13]) or the `EDTSurf` [67] sequential softwares. In [19], the authors present another sequential strategy that struggles with molecules having more than 200 atoms and/or regions of high curvature. In [69], a mesh is built upon the triangulation of an alternative, approximated definition of the solvent-solute interface, defined as a level-set of a sum of Gaussian functions associated with all atoms: no equivalence with the actual molecular surface or accuracy analysis is showed to validate the technique. Finally, another meshing technique is used in `APBS` [1]. It was successfully implemented on parallel architectures [3,4] although partitioning the grid leads to computational overheads: smaller problems, on a coarse mesh, need to be solved by every single process in order to determine the domain partitions, which slows down the procedure and impedes its scalability.

On the contrary, FD methods in a level-set framework allow grid lines and grid cells to be crossed arbitrarily by the interface, where special treatment and vigilant care in the implementation is required. Among all conceptual tools available in the level-set framework, the *reinitialization* [25,45,58,65] plays an essential role, as it allows the calculation of a signed distance to the interface, while handling topological difficulties inherently, in its own mathematical formulation. Moreover, those techniques have proved scalable and capable of capturing highly convoluted surfaces [27,46].

In this paper, we present a fast and scalable algorithm to create an implicit representation of the *Solvent-Excluded Surface* Γ_{SES} (also dubbed *Molecular Surface* in the literature) of a biomolecule using the zero-level set of a function $\overline{\phi}$ and to build a corresponding adaptive Cartesian Octree grid, of aspect ratio 1. The latter is represented as a set of distributed adaptive octree, as implemented in the open-source `p4est` library [11,37]. A two-dimensional illustration of such a grid, *i.e.*, a Quadtree grid, is given in Fig. 1. We note that the algorithms presented in this work have been optimized for the above purpose. When considering the (area of the) *Solvent-Accessible Surface* only, other efficient and optimized approaches may be better suited [29,50].