MIDWEST INTEGRATED CENTER FOR COMPUTATIONAL MATERIALS

# http://miccom-center.org

CONTRACTOR OF

#### Topic: Continuum-Particle Simulation Software (COPSS) Electrostatic Polarization

Presenter: Xikai Jiang, The University of Chicago

2017 Summer School





#### **Continuum-Particle Simulation**



- Continuum simulation (Grid-based) solves Partial Differential Equations on discretized grids or meshes (Finite element method, Finite difference method, *etc.*)
- Particle simulation solves Equation of Motion, *e.g.*, Netwon's 2<sup>nd</sup> law, to evolve positions and velocities of discrete particles (Molecular dynamics, Dissipative particle dynamics, *etc.*)
- Continuum-particle coupling aims to bridge multiple length-scales for complex physics problems (e.g. part of the domain using continuum simulation, other parts use particle-simulation)



#### **COPSS - Polarization**



- Continuum simulation (Grid-based) solves Poisson equation on discretized meshes using Boundary Element Method on the surfaces of colloidal particles
- Particle simulation solves Equation of Motion, *i.e.*, Netwon's 2<sup>nd</sup> law, to evolve positions and velocities of colloidal particles
- Continuum-particle coupling: continuum simulation provides force fields, particle simulation uses the force fields to evolve particles' motion





#### **Dielectric Polarization**



- Dielectric materials (insulator) undergo electric polarization under the influence of applied electric fields
- The charges in dielectric materials don't move freely but only shift slightly from their equilibrium positions in applied electric fields
- Bound charges (induced charges) accumulate on the boundary of the dielectric materials, which ultimately affect force fields acting on dielectric materials and their motions





#### **Dielectric Polarization**

#### 4.2.2 Physical Interpretation of Bound Charge

In the last section we found that the field of a polarized object is identical to the field that would be produced by a certain distribution of "bound charges,"  $\sigma_b$  and  $\rho_b$ . But this conclusion emerged in the course of abstract manipulations on the integral (4.9) and left us with no clue as to the physical meaning of these bound charges. Indeed, some authors give you the impression that bound charges are in some sense "fictitious"—mere bookkeeping devices used to facilitate the calculation of fields. Nothing could be farther from the truth;  $\rho_b$  and  $\sigma_b$  represent *perfectly genuine accumulations of charge*. In this section I'll show you how polarization leads to such

accumulations of charge.

The basic idea is very simple: Suppose we have a long string of dipoles, as shown in Fig. 4.11. Along the line, the head of one effectively cancels the tail of its neighbor, but at the ends there are two charges left over: plus at the right end and

Figure 4.11

MICCo

minus at the left. It is as if we had peeled off an electron at one end and carried it all the way down to the other end, though in fact no single electron made the whole trip—a lot of tiny displacements add up to one large one. We call the net charge at the ends *bound* charge to remind ourselves that it cannot be removed; in a dielectric every electron is permanently attached to a specific atom or molecule. But apart from that, bound charge is no different from any other kind.

To calculate the actual amount of bound charge resulting from a given uniform polarization, examine a "tube" of dielectric parallel to **P**. The dipole moment of the tiny chunk shown in Fig. 4.12 is P(As), where A is the cross-sectional area of the tube and s is the length of the chunk. In terms of the charge at the end, this same dipole moment can be written as. The bound charge that silve would be

5

Introduction to Electrodynamics, D. J. Griffiths, 2<sup>nd</sup> Ed, 1989



## **Induced Charge Computation (ICC)**

A numerical method to solve dielectric polarization problem



 It starts with the Poisson equation for electrostatics subject to boundary conditions for dielectric polarization

$$-\varepsilon_0 \nabla \cdot [\varepsilon(\mathbf{r}) \nabla \psi(\mathbf{r})] = \rho(\mathbf{r})$$

 Poisson equation is transformed to a set of linear system of equations by using boundary-integral theories

$$A\sigma_b = b$$

 Induced charge densities on the dielectric interfaces/boundaries are the unknowns





#### **Induced Charge Computation (ICC)**

Mathematical details on the matrix equation

 $A\sigma_b = b$ 

Matrix-vector multiplication

$$A\sigma_b \equiv \bar{\epsilon}_i \sigma_b + \epsilon_0 \Delta \epsilon_i \mathbf{E}_b \cdot \mathbf{n}$$

 $\epsilon_0$  vacuum permittivity

- $\bar{\epsilon}_i = (\epsilon_i + \epsilon_m)/2$  $\Delta \epsilon_i = \epsilon_m - \epsilon_i$
- Electric field from bound (induced) charges

$$\mathbf{E}_b = \frac{1}{4\pi\epsilon_0} \int_{\partial\Gamma} \frac{\mathbf{r} - \mathbf{r}'}{|\mathbf{r} - \mathbf{r}'|^3} \sigma_b(\mathbf{r}') \, d^2\mathbf{r}'$$

- Right-hand-side  $b = (1 \bar{\epsilon}_i)\sigma_f \epsilon_0 \Delta \epsilon_i \mathbf{E}_f \cdot \mathbf{n}$
- Electric field from free charges

$$\mathbf{E}_{f} = \frac{1}{4\pi\epsilon_{0}} \left( \int_{\partial\Gamma} \frac{\mathbf{r} - \mathbf{r}'}{|\mathbf{r} - \mathbf{r}'|^{3}} \sigma_{f}(\mathbf{r}') d^{2}\mathbf{r}' + \int_{\Omega} \frac{\mathbf{r} - \mathbf{r}'}{|\mathbf{r} - \mathbf{r}'|^{3}} \frac{\rho_{f}(\mathbf{r}')}{\epsilon_{m}} d^{3}\mathbf{r}' \right)$$

Reference: X. Jiang, et al J. Chem. Phys. 145, 064309, 2016



## Solve the ICC Problem

- To solve the matrix equation  $A\sigma_b = b$
- We use Iterative solver GMRES (Generalized Minimal Residual Method)



 Matrix-vector multiplication is the most expensive part in solving the problem

$$A\sigma_b \equiv \bar{\epsilon}_i \sigma_b + \epsilon_0 \Delta \epsilon_i \mathbf{E}_b \cdot \mathbf{n}$$

$$\mathbf{E}_b = \frac{1}{4\pi\epsilon_0} \int_{\partial\Gamma} \frac{\mathbf{r} - \mathbf{r}'}{|\mathbf{r} - \mathbf{r}'|^3} \sigma_b(\mathbf{r}') \, d^2\mathbf{r}'$$





## Solve the ICC Problem

 Matrix-vector multiplication is the most expensive part in solving the problem

$$\mathbf{E}_{b} = \frac{1}{4\pi\epsilon_{0}} \int_{\partial\Gamma} \frac{\mathbf{r} - \mathbf{r}'}{|\mathbf{r} - \mathbf{r}'|^{3}} \sigma_{b}(\mathbf{r}') d^{2}\mathbf{r}'$$
(N)
(N)

- Direct evaluation requires  $O(N^2)$  operations: N discretized nodal points on the boundary mesh,  $E_b$  needs to be evaluated on every nodal points
- Fast Multipole Method (FMM) is used to the computational cost to be in O(N) operations  $f(\mathbf{r}_i) = \sum_{i=1}^{N} K(\mathbf{r}_i, \mathbf{r}_i) \sigma_i$

$$f(\mathbf{r}_i) = \sum_{j=1} K(\mathbf{r}_i, \mathbf{r}_j) \sigma_j$$

Reference: W. Fong et. al., The black-box fast multipole method, J. Compt. Phys., 228, 8712-8725, 2009

 COPSS-Polarization is written on top of ScalFMM (FMM library) and libMesh (parallel finite element and mesh management library)





#### **COPSS-Polarization**



Distribution of induced bound surface charge densities for (a) spherical particles, (b) a spherical particle in a confined geometry, and (c) non-spherical particles. Performance of Induced Charge Computation. (a) Accuracy, (b) efficiency – the O(N) scaling, and (c) parallel scalability.

 COPSS-Polarization houses parallel and efficient solver for electrostatic polarization problem involving multiple dielectric interfaces

MICCoM codes: http://miccomcodes.org/

COPSS-Polarization codes: https://bitbucket.org/COPSS/copss-polarization-public.git



#### **Access COPSS-Polarization**

#### COPSS-Polarization repository on Bitbucket.



#### **Access COPSS-Polarization**

#### Run examples in COPSS-Polarization repository.

COPSS / COPSS-Polarization * +					
( Atlassian, Inc. (US)   https://bitbucket.	org/COPSS/copss-polarization-public/src/e2ac050a2	8088f505620	)3ddc49017929a134	44dd/= 🖾 🤇 🤇 Search	☆ 🖻 🖡 🎓 🛡 🖸 🚺 🚍
Bitbucket Features Pricing				Find a repository Q	⑦ English <b>→ Sign up</b> Log in
COPSS-Polarization-public ACTIONS Clone	MICCOM / COPSS / COPSS-Polarization-public Source COPSS-Polarization-public / examples / sphere / 2_spheres /				
-C Fork	<b>t</b>				
		630 B	2017-05-08	New file names. Class name change from RigidParticles to	DielectricInterfaces.
	■ control.in	428 B	2017-05-08	New file names. Class name change from RigidParticles to	DielectricInterfaces.
	interfaces.in	114 B	2017-05-08	New file names. Class name change from RigidParticles to	DielectricInterfaces.
	i run.py	397 B	2016-10-21	Disable SSE and AVX when install ScalFMM. When it is en	abled, ScalFMM computes wrong
Q Commits	two_sphere_surfaces.jou	1.3 KB	2017-05-08	New file names. Class name change from RigidParticles to	DielectricInterfaces.
Branches	two_sphere_surfaces_12112elem.e	824.9 KB	2016-12-02	Implemented moving particles based on in.params. The init	tial position of all particles in the mesh
Pull requests	two_sphere_surfaces_2.5_1444elem.e	50.6 KB	2016-06-27	file arrangements	
Dipelines NEW	two_sphere_surfaces_2974elem.e	203.5 KB	2016-12-02	Implemented moving particles based on in.params. The init	tial position of all particles in the mesh
Issues	two_sphere_surfaces_722elem.e	50.6 KB	2016-12-02	Implemented moving particles based on in.params. The init	tial position of all particles in the mesh
🕒 Wiki					
Composition Downloads	This example calcualtes the bound surface charge density and body forces for two polarizable spheres.  1. Two input files: interfaces.in: physical parameters for each interface control.in: name and location of input mesh file, and parameters for FMM and convergence  2. Run the example by python run.py It will output the body forces and an output.e for visualization.				
«	3. In the mesh file, each sphere has a subdomain_ID. If you visualize the Exodus file using Paraview, it's called ObjectID.				



# Thanks for your attention.

Q & A.





