

UN METODO MESHFREE PARTICELLARE PER LA RISOLUZIONE DELLE EQUAZIONI DI MAXWELL: SVILUPPO DI UN CODICE IN AMBIENTE GRID

Guido Ala¹, Elisa Francomano², Antonino Spagnuolo¹,
Adele Tortorici², Elena Toscano², Fabio Viola¹

¹Dipartimento di Ingegneria Elettrica, Elettronica e delle Telecomunicazioni

²Dipartimento di Ingegneria Informatica

Università degli studi di Palermo - Viale delle Scienze, 90128 Palermo

La crescente necessità di risolvere complessi problemi elettromagnetici ha determinato un massiccio impiego di simulazioni al calcolatore basate su metodi numerici sempre più avanzati. D'altra parte, la generazione iniziale della *mesh* o della griglia, in presenza di geometrie complesse o irregolari può essere di ardua implementazione e può richiedere l'impiego di tecniche localmente adattative che necessitano di risorse computazionali aggiuntive molto gravose. Tali aspetti negativi si aggravano in presenza di condizioni diffuse di non omogeneità, superfici libere, contorni deformabili; inoltre tali metodi mal si prestano a simulazioni di problemi fisici in cui gli oggetti presentano caratteristiche di *particelle* (interazione stellare in astrofisica, moto degli atomi, etc.)

Le tematiche sopra riportate vengono affrontate con metodi computazionali denominati *meshfree*; l'idea chiave è quella di ottenere soluzioni numeriche per equazioni integrali e/o differenziali con le opportune condizioni al contorno, utilizzando un insieme di *particelle* (o nodi) arbitrariamente distribuite. Tra i metodi *meshfree* particellari, quello denominato “*Smoothed Particle Hydrodynamics*” (SPH) [1], intrinsecamente adattativo, sviluppato inizialmente in ambito astrofisico, e successivamente esteso a problemi di fluidodinamica, biologia molecolare, etc., viene applicato, nella ricerca corrente, all'analisi elettromagnetica (SPEM) [2]. La seguente procedura sintetizza i passi fondamentali per l'applicazione del metodo:

- 1) discretizzazione del dominio di definizione mediante un set di particelle distribuite arbitrariamente; nessuna connettività è richiesta per tali particelle;
- 2) approssimazione delle funzioni di campo mediante il metodo della rappresentazione integrale (*kernel approximation*);
- 3) approssimazione dell'integrale mediante sommatorie su tutti i valori assunti in corrispondenza delle particelle vicine che ricadono nel dominio di supporto (*particle approximation*);
- 4) la *particle approximation* viene ripetuta per tutti i termini che contengono funzioni spaziali di campo, in modo da ottenere un set di equazioni differenziali ordinarie in forma discreta dipendenti solo dal tempo;
- 5) tale set di equazioni temporali viene quindi risolto usando uno schema esplicito alle differenze finite.

Il passaggio fondamentale del procedimento è costituito dalla rappresentazione integrale della funzione di campo e delle sue derivate. Si introduce a tal fine una funzione detta *smoothing kernel*, e di seguito indicata con $W(\mathbf{x}-\mathbf{x}',h)$, la quale, moltiplicata per la funzione di campo ed integrata sul supporto Ω fornisce la rappresentazione integrale cercata:

$$\langle f(\mathbf{x}) \rangle = \int_{\Omega} f(\mathbf{x}') W(\mathbf{x}-\mathbf{x}',h) d\mathbf{x}' \quad (1)$$

essendo $\mathbf{x}$ e $\mathbf{x}'$ vettori spaziali di posizione ed h è la cosiddetta *smoothing length* che definisce l'area di influenza della funzione W ed ha un peso rilevante nell'accuratezza della soluzione. Affinché la (1) costituisca una efficiente rappresentazione di $f(\mathbf{x})$, la funzione $W(\mathbf{x}-\mathbf{x}',h)$ deve godere di specifiche proprietà, in modo da approssimare la funzione delta di Dirac. Uno dei *kernel* che si adatta allo scopo è quello gaussiano. Altro kernel di interesse è quello ottenibile con le funzioni *B-spline* a supporto compatto. Partendo dalla (1) è possibile dimostrare che la rappresentazione integrale delle derivate della funzione di campo consente di trasferire l'operatore di derivazione dalla funzione al *kernel* interpolante. Ciò determina risultati accurati se il dominio di supporto dello *smoothing kernel* non interseca il contorno del dominio del problema in esame. Se invece ciò avviene occorre introdurre delle modifiche alla formulazione per tenere conto del troncamento operato.

Nell'ambito del progetto PI2S2 (Progetto per l'Implementazione e lo Sviluppo di una e-Infrastruttura in Sicilia basata sul paradigma della GRID), gestito dal Consorzio COMETA (COnsorzio Multi Ente per la promozione e l'adozione di Tecnologie di calcolo Avanzato), si è implementato un codice di calcolo distribuito basato sul metodo SPEM. Il codice in C++ prevede l'impiego delle librerie MPI (Message Passing Interface), scritte appositamente per facilitare l'implementazione multiprocessore di codici di calcolo. Il dominio del problema è stato suddiviso tra i diversi processori. Ogni processore ha quindi in memoria soltanto una parte del dominio, più la striscia di frontiera (in giallo in figura) in comune con gli altri processi. A titolo di esempio viene riportato in figura il caso di 4 processori (P0, P1, P2, P3):

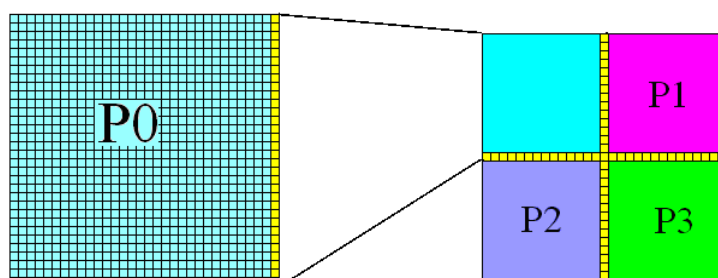


Figura 1 - Suddivisione del dominio tra quattro processori

Dopo la fase di calcolo di uno dei campi, i dati delle particelle della striscia del dominio in comune vengono trasferiti alla frontiera del processo adiacente. Procedendo in questo modo si ha un bilanciamento delle operazioni (ogni processo deve eseguire gli stessi calcoli) e delle comunicazioni (ogni processo deve trasferire e ricevere la stessa quantità di dati). Si riporta un confronto dei tempi di calcolo rispetto ad una simulazione su un solo processore riferita ad un caso di propagazione nello spazio libero bidimensionale:

Tempi di calcolo (ms)	Monoprocessore	Parallelo con 4 processori
Calcolo della derivata del kernel	19217	5730
Durata del singolo step temporale	30	180
Trasferimento dati tra i processi	Assente	40 x 2 =80
Totale	19247	5990

Tabella 1 – Confronto tra i tempi di calcolo distribuito e seriale

Bibliografia

- [1] G. R. Liu, Smoothed Particle Hydrodynamics, World Scientific, 2003.
- [2] G. Ala, F. Viola, E. Francomano, A. Tortorici, E. Toscano: "A Smoothed Particle Interpolation Scheme for Transient Electromagnetic Simulation" . IEEE Transactions on Magnetics, Volume 42, number 4, 2006.