UN CODICE 3D PARTICLE IN CELL FEM PARALLELO PER LO STUDIO DELLA DINAMICA DI FASCI DI PARTICELLE CARICHE

S. Coco, A. Laudani, G. Pollicino, P. Tirrò

DIEES, Università di Catania
Viale A. Doria 6, 95125, Catania (CT)

La dinamica delle particelle cariche soggette ad un campo elettromagnetico in assenza di collisioni può essere descritta per mezzo dell'equazione di Vlasov per la funzione di distribuzione f(t, x, p), definita nello spazio di fase (x, **p**) al tempo t

$$\frac{\partial f}{\partial t} + \frac{\mathbf{p}}{m} \cdot \nabla_x f + q(\mathbf{E} + \frac{\mathbf{p}}{m} \times \mathbf{B}) \cdot \nabla_p f = 0$$

dove **p** è il momento delle particelle, m e q sono rispettivamente la massa della particelle e la carica. Il campo elettrico **E** e il campo magnetico **B** soddisfano l'equazione di Maxwell

$$\rho(t,\mathbf{x}) = q \int_{R_p} f(t,\mathbf{x},\mathbf{p}) d\mathbf{p} \quad \mathbf{J}(t,\mathbf{x}) = q \int_{R_p} \frac{\mathbf{p}}{m} f(t,\mathbf{x},\mathbf{p}) d\mathbf{p}$$

e la carica e la densità di corrente sono le sorgenti di campo. L'equazione di Vlasov con l'equazione di Maxwell costituiscono un sistema di equazioni accoppiate, chiamato sistema di Vlasov-Maxwell. La soluzione di questo problema accoppaito può essere effettuata usando l'approccio Particle In Cell (PIC) rappresentando il fascio con un numero ragionevole di macro-paraticelle puntiformi. Nel processo di soluzione l'iterazione con i campi self-consistent e i campi esterni applicati è ottenuta tramite un appropriato schema computazionale, che ad ogni step prevede due fasi:

1. l'integrazione delle traiettorie, la distribuzione della carica spaziale e della densità di corrente.
2. la soluzione delle equazioni del campo.

Per risolvere l'equazione di campo è conveniente adottare un approccio ad elementi finiti (FE), adatto alla descrizione di geometrie complesse 3D dei dispositivi elettronici a vuoto. Purtroppo la soluzione numerica implica la gestione e l'elaborazione di una quantità enorme di dati, per cui si rende necessaria la paralellizzazione del problema.

In questa memoria viene presentato un codice 3D FE PIC time-dependent sviluppato in un ambiente computazionale distribuito su larga scala (la griglia computationale), in cui la fase 1 viene realizzata da diversi processori. Infatti sebbene la soluzione delle equazione de campo sia uno dei principali task del PIC code, il suo tempo di esecuzione è inferiore del 10-15% del tempo totale di simulazione, mentre il collo di bottiglia è il tracciamento delle traiettorie. Per ridurre il computing time si divide l'integrazione delle macro-particelle fra i vari processori, utilizzando un bilanciamento dinamico del carico. Per fare ciò si affida la gestione del processo ad un processore master, che ad ogni timestep calcola il potenziale elettrico, conta le particelle e le assegna ai vari processori slave per il tracciamento. Ad ogni processore è assegnato un numero di particelle variabile in relazione ad una soglia funzione del numero di macroparticelle presenti e del tempo di esecuzione della precedente iterazione. Così facendo si garantisce un ridotto interprocess comunication. Ovviamente dopo un certo numero di passi, il numero di particelle sarà abbastanza grande da richiedere l'utilizzo di tutti i processori. In ogni caso poiché ogni processore gestisce il proprio set di particelle non occorre lo scambio dei dati relativo alle traiettorie, fin tanto che le particelle non devono essere ridistribuite fra i vari processori (load balancing), evento che si verifica se il numero di

particelle è maggiore della soglia. Il numero di processori paralleli per ottenere lo speed-up ottimale dipende dal numero di particelle coinvolte nella simulazione. Si è affrontata tale problematica facendo uso di un algoritmo euristico adattativo che evita un eccessivo interprocess comunication. In particolare il processore master stabilisce il numero ottimo di processori slave, dove viene lanciata la procedura di tracciamento delle particelle in accordo con il numero di particelle e al tempo di simulazione risultante nel precedente time step. L'algoritmo messo a punto è di natura euristica a causa della variazioni ad ogni time step del numero di particelle interne al dominio, il cui valore non è stimabile tramite una legge deterministica (inserimento e estrazione); per cui durante la simulazione anche processori usati saranno di numero variabile.

Di seguito si presenta un test relativo all'espansione di un fascio laminare all'interno di un tubo cilindrico (spread beam test). I risultati ottenuti sono in accordo con quelli analitici disponibili in letteratura. In condizioni steady-state si trovano nel dominio circa 2M di particelle. Per quanto riguarda le performance del test, è stato possibile stabilire che:

1 il numero minimo di particelle sopra il quale è effettuato il primo split è di 100k, ed il numero ottimale medio di particelle è di 120K (quando vengono usati tutti e 16 i processori disponibili)

2 usando il bilanciamento adattativo del carico l'aumento della velocità complessivo è di circa 10-20% rispetto ad un numero fisso di processori.

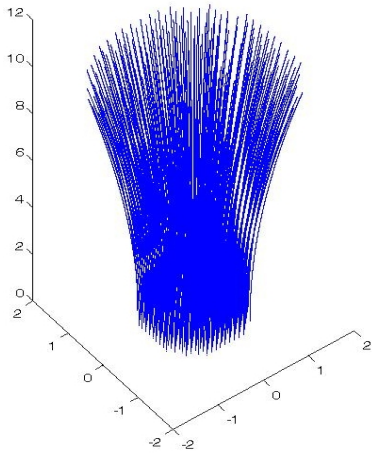

Fig. 1 - Spread beam test

Referenze

[1] S. Coco, A. Laudani, G. Pollicino, "A 3-D FE Particle-in-Cell Parallel code with adaptive load balancing", *COMPUMAG 2009*, Florianopolis, Brasile, 22-16 Novembre 2009

[2] E. Gjonaj, "Particle-In-Cell Simulations: Techniques and Applications", *OCPEM-Workshop*, Napoli, September 22-23, 2003

[3] S. Coco, A. Laudani, G. Pollicino, "Distributed computing finite element electromagnetic analysis of traveling wave tubes", *COMPEL*. vol. 27, pp. 1326-1334 (2008).

[4] F. Wolfheimer, E. Gjonaj, T. Weiland, "A Parallel 3D Particle In Cell (PIC) with Dynamic Load Balancing" *Nuclear Instruments and Methods in Physics Research (NIM)*, Vol. 558, pp. 202-204, (2006)

[5] Hong-Yu Wang, Wei Jiang, You-Nian Wang, "Parallelization and optimization of electrostatic Particle-in-Cell/Monte-Carlo Coupled codes as applied to RF discharges", *Computer Physics Communications* 180 1305–1314(2009)