CRS4

Centre for Advanced Studies,

Research and Development in Sardinia

Uta - (CA)

FEMME:
Fluid-dynamics Engineering Montecarlo Method Environment

Date: June 14^{th} 2001

by:

A. Varone

Computational Fluid Dynamics Area

Abstract

A new software platform based on the particle transport and montecarlo techniques is described. The code, named FEMME, is a new portable enviromment that's naturally integrated whith the in house fluid dynamics RANS code of CRS4 (KARALIS). This code permits to simulate physical problems associate to the fluid motion that depends on the lagrangian displacement of *particles* in the fluid. Pratically developed an hybrid algorithm particle-finite-volume to join the Reynolds averaged Navier Stokes equations and Montecarlo transport equations,to solve particular and complex fluid dynamics problems joined to reactive or turbulent flows.

The first application carried out is concerned with the Rubbia's Project 242 [1]. Other filed of application may be related to combustion problems, to improvement of turbulence modelling, as well as to the transport of passive scalars in environnemental and geophysical applications.

Contents

1 Introduzione

FEMME (Fluid-dynamics Engineering Montecarlo Method Environment) é un nuovo codice nativo del CRS4 sviluppato per poter sfruttare le metodologie Montecarlo nello studio della dinamica dei fluidi. Le grandi potenzialitá di una macchina parallela possono in questo modo essere sfruttate ampiamente per effettuare un approccio statisico alla risoluzione di problematiche fisiche di svariata natura.

Ogni qual volta ci si trova a dover effettuare simulazioni che coinvolgono un elevato numero di *eventi* o *particelle* la possibilitá di utilizzare uno studio dinamico statistico tramite l'utilizzo contemporaneo di diversi processori, rende il metodo Montecarlo un ottimo candidato per lo sviluppo di nuovi codici computazionali. Difatti le possibiltá offerte dall'approccio statisco ad un problema fisico di questo tipo, sono perfettamente portabili su una macchina parallela nella quale ogni processore lavora indipendentemente dagli altri. Solo alla fine dei conti, i risultati dei singoli processori vengono raccolti ed analizzati i risultati statistici. É stato perció sviluppato un codice Montecarlo che puó interagire direttamente con i codici di fluidodinamica nativi del CRS4.

Sulla base delle esperienze sviluppate nell'ambito del progetto P242 [1] in cui per la prima volta sono state interfacciate al codice del gruppo di fluidodinamica KARALIS [2] dei risultati provenienti da simulazioni Montecarlo (codici FF_HEAT e PHOTON [4]) é stata sviluppata una piattaforma computazionale che permette di utilizzare le tecniche Montecarlo utilizzando direttamente i risultati delle simulazioni fluido dinamiche senza dover passare attraverso trasformazioni delle strutture dati [5].

Tramite questo nuovo codice possono esssere affrontate, contemporaneamente alla risoluzione delle equazioni di Navier-Stokes prodotta dal codice RANS nativo del CRS4, alcune problematiche fisiche legate direttamente al trasporto lagrangiano di una qualsiasi grandezza. Nel caso particolare il primo obiettivo é stato quello di simulare il riscaldamento dovuto al trasporto dei frammenti emessi dalla reazione di un materiale fissile (Americio 242) distribuito sulle pareti cilindriche della camera di combustione di quello che risulta essere un elemento base per il razzo a propulsione nucleare [1].

2 La struttura del codice

La semplice struttura del codice **FEMME** é riportata nella Figura 1. Il diagramma a blocchi rappresenta schematicamente le operazioni compiute dal codice. Per prima cosa vengono letti i dati in entrata relativi alla griglia ed ai campi interessati. Il formato dei dati risulta essere del tutto identico a quello generato in uscita dal codice

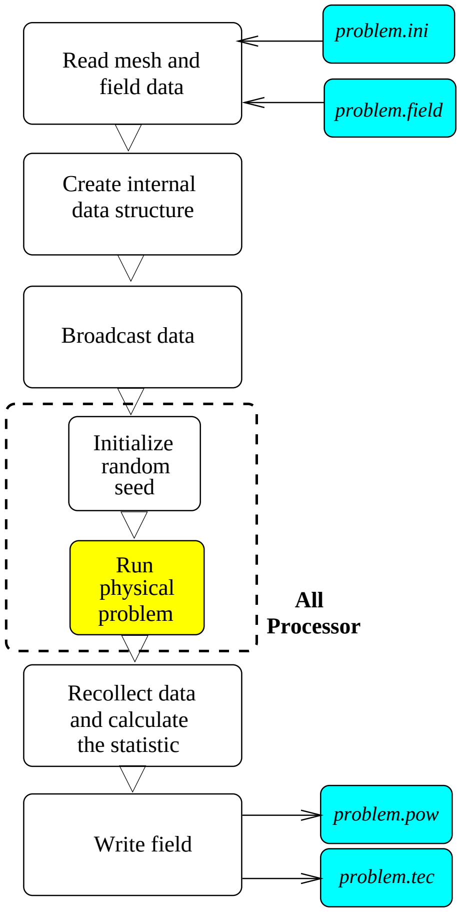

Figure 1: Diagramma a blocchi della piattaforma FEMME

Karalis. Per un dato problema in esame, *problem*, i files di input risultano quindi essere del tipo *problem.msh* contenente le informazioni sulla griglia e *problem.sol* dove sono allocate i valori dei campi (scalari e/o vettoriali) da utilizzare per la simulazione. Essendo il codice KARALIS di tipo multiblocco ad ogni blocco in cui vine suddiviso il dominio computazionale sono associati due files relativi alla griglia (*.msh*) ed ai campi interessati (*.sol*).

Inizialmente quindi il codice Montecarlo legge, come il codice KARALIS, il file PREFIX contenente il nome del problema in esame. Di seguito viene richiesto di indicare quanti dei blocchi in cui é diviso il dominio si vogliono leggere. Questo perché non sempre tutti i blocchi sono necessari nella simulazione montecarlo.In altre parole la modellistica fisica richiesta puó in generale interessare solo una parte dell'intero dominio computazionale. Come descritto in seguito, nel caso del P242, il rilascio dei frammneti di fissione avviene solo in alcune zone della camera di combustione e non in tutto il dominio della simulazione fluidodinamica. Una volta lette le griglie dei vari domini viene definita una sola griglia computazionale su cui avvengono le simulazioni montecarlo. Nella piattaforma **FEMME** difatti, non avviene una frammentazione del dominio computazionale in quanto sono i diversi eventi a venire ripartiti sui diversi processori. La stessa griglia viene quindi passata a tutti i processori durante l'operazione di broadcast, per ognuno di essi avviene l'inizializzazione della routine di generazione dei numeri random. Ogni processore quindi simula sulla stessa griglia un numero identico di *eventi* generati casualmente con differenti inizializzazioni. È a questo punto che interviene nel codice la modellizazzione del problema fisico che si intende descrivere. Il moto degli eventi viene comunque descritto dalle routines di *path finder* del codice generale, mentre le interazioni con i diversi campi fluidodinamici congelati (pressione, densitá, campi di velocitá etc...) sono simulate dalle routines costruite appositamente per il problema in esame. In questo report viene citato ad esempio e descritto in maggiore dettaglio nei paragrafi seguenti lo sviluppo del codice montecarlo atto a simulare il riscaldamento del gas all'interno della camera di combustione. I risultati di ogni singlo processore vengono quindi ridotti (messi insieme) sul singolo processore, a questo punto possono essere effettuate le appropriate operazioni di statistica che ci forniscono i risultati della simulazione sulle singole celle del dominio. I risultati in uscita sono associati alle celle (o ai vertici) del dominio della siimulazione fluidodinamica e contengono le informazioni relative ai campi e/o agli scalari fisiche in esame. Come vedremo in seguito, nel caso particolare del codice di riscaldamento, l'output della simulazione montecarlo é costituito da un campo di densitá di potenza associato alle celle dovuto alla cessione di calore da parte dei frammenti di fissione nell'attraversamento del gas. I campi cosí ottenuti vengono quindi riscritti nel formato del codice fluidodinamico, su diversi files associati a ciascun blocco (*.pow*), e in un formato che ne permetta la visualizzazione (*.tec*) tramite i codici commerciali standard.

A grandi linee questa é la struttura di funzionamento del codice FEMME, maggiori dettagli come ad esempio la struttura delle routines di ray tracing sono fornte nel capitolo seguente dedicato all'implementazione del codice atto a simulare il riscaldamento del gas.

3 Il codice per il riscaldamento

Nell'ambito del progetto denominato **ASI242** [1] è stato sviluppato al CRS4 un nuovo codice integrato che permette la simulazione dei processi termofluidodinamici all'interno della camera di combustione di un razzo di nuova concezione.

Il riscaldamento del gas (idrogeno in tutte le simulazioni), avviene tramite la cessione di calore da parte degli ioni carichi (frammenti) emessi dalla reazione di fissione del materiale distibuito sulla parete (Americio 242 nei test preliminari). Questo consente al gas di raggiungere temperature molto elevate (circa $10000°K$) altrimenti

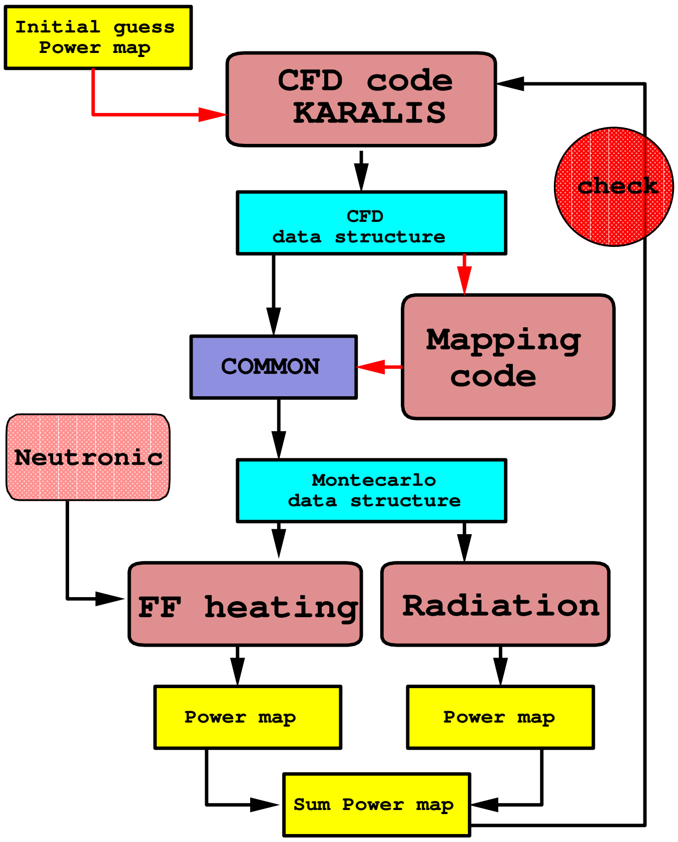

Figure 2: Il sistema integrato **SUPER_ROCKET**

non raggiungibili nei razzi a propulsione convenzionale (ad es. chimici ed anche nucleari con trasmissione del calore convenzionale tramite convezione). Data l'elevata temperatura nel cuore della camera di combustione e gli elevati gradienti termici, diventa fondamentale lo studio del rilascio di calore sotto forma di radiazione e la conseguente ridistrubuzione del calore all'interno della camera.

Per studiare l'effetto contemporaneo di tutti questi fenomeni è stato implementato al CRS4 un codice integrato denominato SUPER_ROCKET il quale sfruttando le peculiaritá di codici diversi tra di loro ha permesso di ottenere una simulazione completa della fisica all'interno della camera di combustione [3]. In particolare l'attenzione é stata rivolta all'interfaccia tra i codici fluidodinamici nativi del CRS4 (KARALIS) ed i codici di Montecarlo di nuova generazione sviluppati all'interno del progetto P242 [4]).

Il problema é sorto nel momento in cui per simulare la fluidodinamica del gas all'interno della camera, si é reso necessario affiancare al codice KARALIS dei codici sviluppati appositamente per simulare la fisica del riscaldamento del gas da parte dei frammenti di fissione provenienti dalle pareti e la ridistribuzione del calore stesso ad opera dei processi di radiazione ed assorbimento.

Questi codici, denominati FF_HEAT e PHOTON rispettivamente per il trasporto dei frammenti di fissione e per il calcolo degli scambi termici dovuti all'emissione e all'assorbimento delle radiazioni elettromagnetiche, si basano su approcci matematici differenti da quelle su cui si svluppa il codice fluidodinamico. Difatti se nel secondo caso lo schema di simulazione è di tipo euleriano, in cui vengono esaminati e trattati i flussi delle grandezze fisiche in gioco attreaverso le superfici delle celle elementari, nel caso dei codici FF_HEAT e PHOTON l'approccio é di tipo Lagrangiano, vengono quindi seguite le particelle (frammenti o fotoni a seconda dei casi) nella loro evoluzione dinamica. In questo tipo di schema l'approccio al problema risulta essere di tipo statistico, difatti vengono simulate (lanciate) un elevato numero di particelle tramite le quali si ottiene, utuilizzando gli appropraiti strumenti statistici, il risultato atteso. Si parla quindi in questo caso di approccio Montecarlo.

In particolare nei due casi, Euleriano e Lagrangiano, il tipo di griglie utilizzate risulta essere completamente diverso: nel primo caso le griglie sono costruite in maniera tale da essere il piú conformi possibile alla fisica attesa (in particolare nel caso del razzo assialsimmetrica) mentre nel secondo caso, per motivi puramente statistici, le griglie risultano essere puramente cartesiane.

Lo schema di funzionamento del codice integrato SUPER_ROCKET é illustrato in figura 2. Come si nota dalla figura la separazione tra i diversi codici risulta netta. Ogni operazione di aggiornamento dei campi di densitá di potenza ceduta al gas dal passaggio dei frammenti di fissione, risulta rallentata da pesanti operazioni di trasformazione dei dati dalle griglie relative ai codici Mntecarlo (cartesiane omogenee) a quelle su cui é costruito il dominio computazionale del codice KARALIS (non omogenee e adattate alla geometria). Il diverso ambiente sul quale erano sviluppati i due codici, sia dal punto di vista metodologico (Euleriano Lagrangiano) sia per quanto riguarda la diversa natura delle griglie computazionali portava ad un enorme dispendio sia di tempo che di memoria computazionale.

Con la creazione della nuova piattaforma computazionale FEMME l'integrazione dei due sistemi é totale. Il software difatti é stato creato proprio per questa ragione: utilizzare le tecniche montecarlo sulle strutture dati sulle quali vengono effettuate le simulazioni fluidodinamiche.

Nel caso particolare del codice di riscaldamento, su geometrie assialsimmetriche, le migliorie introdotte hanno portato a due sostanziali risultati.

Innanzitutto avere dei codici piú rapidi, difatti se precedentemente la geometria cartesiana sulla quale erano sviluppate le griglie dei codici montecarlo riempiva tutto lo spazio a disposizione del cilindro rappresentante la camera di combustione, nella nuova rappresentazione assialsimmetrica le griglie sono esattamente coincidenti con quelle del codice fluidodinamico e possono quindi, sfruttando le proprietá di simmetria del problema, lavorare su settori circolari n-volte piú piccoli del cilindro originario.

Il secondo risultato dipende dal fatto che le griglie risultano identiche. In questo modo, quindi non risulta piú necessario lavorare con griglie non coincidenti e vengono quindi a cadere i lunghi tempi morti necessari alla trasformazione dei risultati per passarli da una griglia all'altra. Inoltre, in questo modo i codici risultano perfettamente integrati e congruenti rendendo quasi il codice di riscaldamento del gas una routine del codice di fluidodinamica. Inoltre in questo modo il numero effettivo di celle su cui avviene la simulazione montecarlo é sensibilmente ridotto. Infatti se nel caso delle griglie cartesiane omogenee il numero di celle identiche utilizzato nelle simulazioni montecarlo era strettamente vincolato all'ordine di grandezza del volume della piú piccola cella utilizzata nelle simulazioni fluidodinamiche, in una configurazione in cui si mantiene la stessa struttura di griglia il numero di celle rimane lo stesso nei due codici. Si é potuti passare quindi da un numero di celle pari a 8 miloni ($200X200X200$) dei codici montecarlo della piattaforma SUPER_ROCKET, ad un numero che rimane al di sotto delle 10000 unitá nel caso della nuova piattaforma con una sensibile riduzione dei tempi di calcolo.

Quindi see da un punto di vista dell'algoritmo di ray-tracing (ricerca della distanza percorsa da una particella all'interno delle singole celle (vedi Sezione 3.2) si ha un aumento delle operazioni da compiere, da un punto di vista del dominio il guadagno sul tempo computazionale effettivo é talmente elevato dal rendere trascurabili le maggiori operazioni.

3.1 Griglie e strutture dati

Come detto in precedenza la definizione delle griglie sulle quali é sviluppato il software FEMME é del tutto identica a quella utilizzata nel codice KARALIS. Piú precisamente, le griglie dei codici FEMME vengono mutuate integralmente da quelle del codice di fluidodiamica. Nel caso particolare del razzo vista la particolare configurazione della camera di combustione, di forma cilindrica, le simulazioni RANS avvengono su una geometria che sfrutta e condizioni di simmetria assiale del problema in esame. In questo caso la piattaforma **FEMME** é in grado di leggere le griglie di KARALIS e ricrearle identicamente. In questo modo il passaggio dei dati fisici tra i due codici é immediato senza bisogno di nessuna operazione di trasformazione.

In figura 3 é riportata la griglia computazionale su cui avvengono le simulazioni fluidodinamiche. Data la natura assialsimmetrica del problema (la camera di combustione é di forma cilindrica e tutte le condizioni di input sono a simmetria assiale) l'effettiva griglia di calcolo é sviluppata su un solo sttore cicolare di ampiezza $\frac{\pi}{16}$. Nella figura viene rappresentata a sinistra la griglia completa. Le misure sono di $0.2m$ per il ragggio e di $2.5m$ per la effettiva altezza della camera di conbustione (escluso l'ugello). A destra in alto ed in basso sono rappresentate rispettivamente

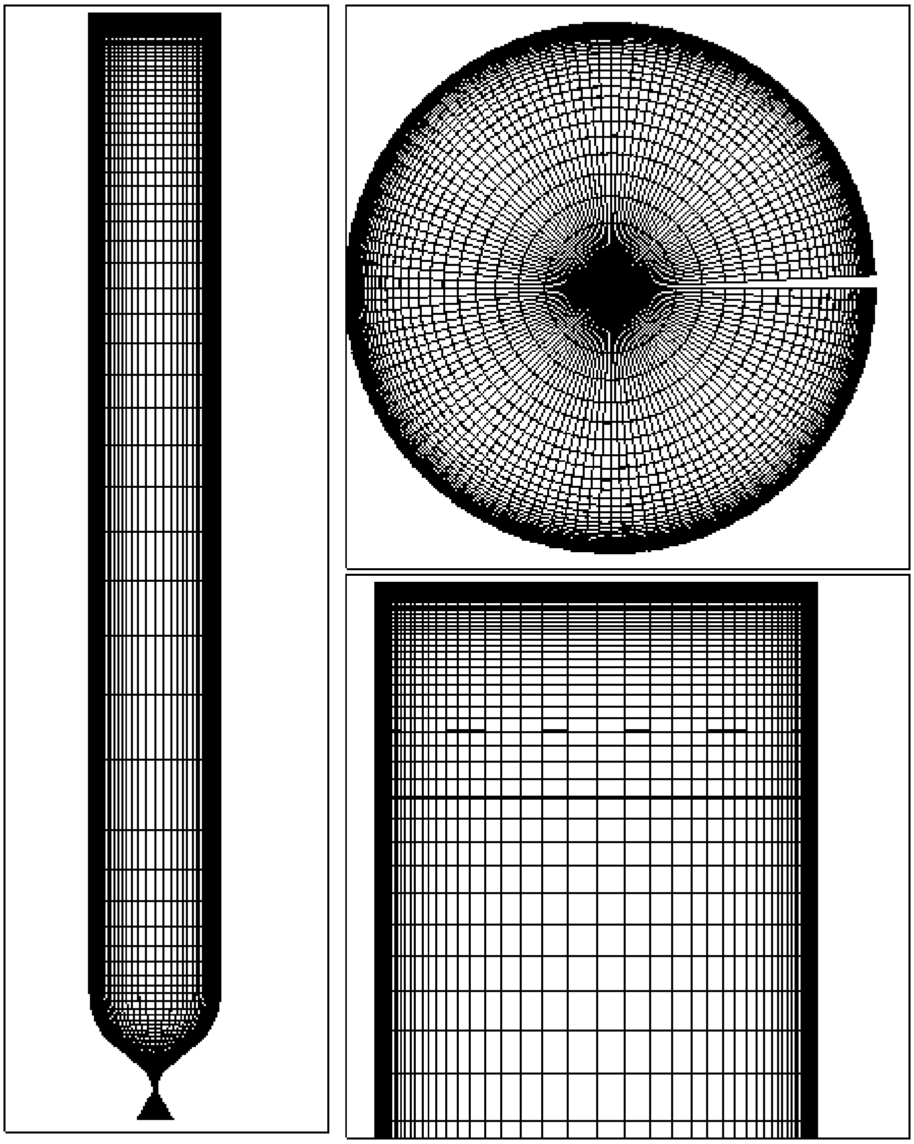

Figure 3: La griglia del razzo.

una sezione ed un ingrandimento della parte superiore della camera, in modo da evidenziare il raffinamento della griglia in vicinanza delle pareti. Nel caso della sezione la griglia viene riprodotta identicamente per 16 volte.

Le griglie computazionali vengono salvate dal codice KARALIS nei files *.msh* tramite le coordinate dei piani in sucessione per ognuno dei blocchi in cui é suddivisa la griglia. Questi files vengono letti direttamente dal codice FEMME che utilizza soltanto i blocchi necessari al calcolo Montecarlo. Nel caso particolare della camera di combustione, in cui i frammenti di fissione vengono emessi solo nella parte superiore delle superfici interne, non tutti i blocchi della griglia fluidodinamica sono necessari. Quindi vengono utilizzati solo i blocchi della camera cilindrica ad una certa altezza al di sopra dell'ugello (nei casi test 0.5 metri a parti dall'inizio della curvatura dell'ugello!).

La griglia della struttura montecarlo viene quindi costruita in questo caso utilizzando le coordinate dei vertici delle celle fluidodinamiche trasformate in modo da costruire la matrice dei piani $\rho\ \theta\ Z$ che costituiscono la griglia cilindrica dove avviene il riscaldamento del gas da parte dei frammenti di fissione. I dati così organizzati vengono inseriti in una matrice:

$$PLANES(3, N\_MAX\_PLANES)$$

la cui dimensione massima N_MAX_PLANES é definita in base al numero massimo di piani in una direzione ($N_Z = 80$ nel caso in considerazione).

Con una griglia strutturata i questo modo risultano quindi definite univocamente le celle, la cui terna di identificazione I, J, K varia negli intervalli $1 \cdots N_\rho$,$1 \cdots N_\theta$ e $1 \cdots N_Z$. La possibilitá di conservare una griglia strutturata permette di mantenere un algoritmo di propagazione degli eventi attreaverso le celle a basso costo computazionale.

3.2 Il nuovo algoritmo di *ray tracing*

L'efficenza del codice Montecarlo dipende in maniera cruciale dalla rapiditá di funzionamento dell'algoritmo di propagazione rettilinea degli *eventi* attraverso la griglia.

Dato infatti l'elevato numero di *eventi* che bisogna simulare per poter ottenere una buona convergenza statistica l'operazione di attraversamento delle celle e la misurazione delle distanze compiute da ciascun evento durante l'attrversamento delle celle risultano le operazioni maggiormente utilizzate dal codice.

Se in una griglia cartesiana ed ortogonale queste operazioni risultavano semplici e rapide, nel caso di griglie piú complesse la struttura delle routines ed i conseguenti tempi computazionali spesi all'interno di ciascuna di esse, possono appesantire enormemente le operazioni, rischiando di inficiare la validitá dell'algoritmo montecarlo.

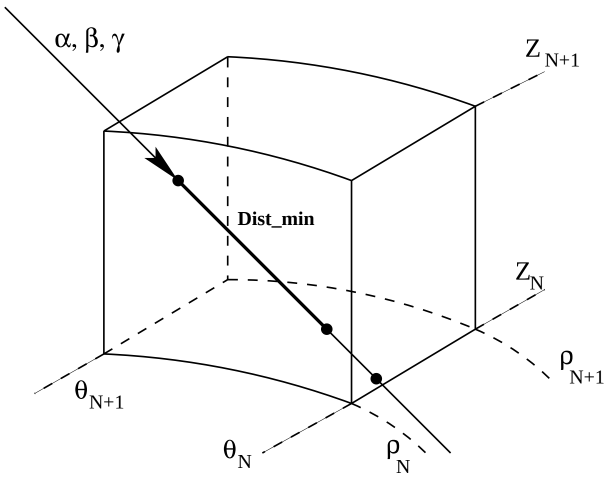

Figure 4: La ricerca della minima distanza.

Nelle griglie cartesiane ortogonali [4] infatti tutte le celle della griglia (voxels) risultano essere tra loro identiche, per cui una volta definite le coordinate del punto di ingresso nel voxel ed i coseni direttori della direzione di propagazione, le operazioni da compiere per identificare il punto d'uscita e conseguentemente la distanza percorsa all'interno del voxel risultano immediate. Nel caso di geometrie curvilinee invece viene persa l'autosomiglianza tra le celle, essendo nel caso in esame la griglia identica a quella fluidodinamica le in cui dimensioni delle celle risultano completamente diverse tra loro. Inoltre il fatto di lavorare in geometrie curvilinee aggiunge una ulteriore complicazione all'algoritmo di ricerca del punto di intersezione tra la direzione di propagazone e le superfici definite dai piani. Il cuore della procedura di propagazione é costituito dalla routine HOW_FAR. Una volta che gli *eventi* associati ai frammenti di fissione (vedi appendice A) vengono generati sulla superficie del cilindro con una data direzione di propagazione, viene assegnata la terna di numeri corrispondente alla cella in cui il frammento é contenuto. A questo punto viene chiamata la routine HOW_FAR che in uscita fornisce sia la distanza percorsa dal frammento in quella cella sia gli identificatori del cammino nella griglia ID_I, ID_J, e ID_K per determinare la terna corrispondente alla nuova cella.

Se nel caso di griglie cartesiane ortogonali l'autosomiglianza tra i voxels permetteva di lavorare con una routine del tutto generale (ogni cella poteva venir traslata nel voxel fondamentale) nel caso di griglie in geometrie complesse é necessario che nella routine siano contenute le informazioni relative ai piani (la matrice $PLANES$). Soltanto in questo modo infatti, possono essere identificate le intersezioni lungo la traiettoria dal punto di ingresso nella cella con i piani (in questo caso il termine piani risulta generico e sta ad indicare anche i cilindri a ρ costante) e calcolare la distanza minima. In Figura 4 viene rappresentato schematicamente l'algoritmo di attraversamento di una cella. Una volta identificati i piani piú vicini lugo la traiettoria α, β, γ e calcolate le diverse intersezioni viene identificato il piano a distanza minima (**dist_min**) e di conseguenza la terna identificatrice della nuova cella. L'algoritmo risulta leggermente piú macchinoso rispetto a quello per le celle ortogonali cartesiane, ma non al punto da inficiare la validitá del codice.

In questo modo, una volta determinata la distanza percorsa dal frammento di fissione nella cella, note le proprietá fisiche del gas nella stessa si puó determinare la quantitá di calore ceduta dallo ione al gas.

Ogni ione viene quindi inizializzato sulla parete, attingendo dal database fornito dal cern , con una certa energia ed una direzione di propagazione. Il suo percorso attraverso le celle viene quindi seguito tramite la funzione di *ray tracing* l'energia che esso perde va ad accumularsi nelle celle e la sua evoluzione viene seguita fino a quando non riesce dalle pareti della camera o esaurisce completamente la sua energia. La simulazione ha termine quando tutti i frammenti di fissione definiti inizialmente vengono *sparati*, a questo punto si eseguono le operazione di media statitica sulla matrice delle energie ceduta ad ogni cella. La matrice delle densitá di potenza rielaborate statisticamente e distribuite sulle singole celle é il risultato in uscita dalla simulazione montecarlo ed utilizzato come dato in ingresso dalle simulazioni fluidodinamiche.

3.3 Risultati

*** PARAMETERS ***		
Pression of the gas:		3.9843 atm
Thickness of the fuel (Am242*) layer:		.3 E-05 m
Radius of cylinder supporting the fuel (Am242*)		.2 m
Height of cylinder supporting the fuel (Am242*)		2.5 m
Lattice resolution	(x):	33
	(y):	3
	(z):	81
Active area of Am242*:		3.77 m^2
Power density on Am242*:		200. W/cm^2
Gross Engine thermal power:		7.5398 $MWatt$
No. of FF histories:		100000000
Number of parallel processors:		8

Table 1: I parametri di ingresso.

La versione della piattaforma **FEMME** costruita per simulare il riscaldamento del gas all'interno della camera di combustione é stata sviluppata e applicata a diversi casi

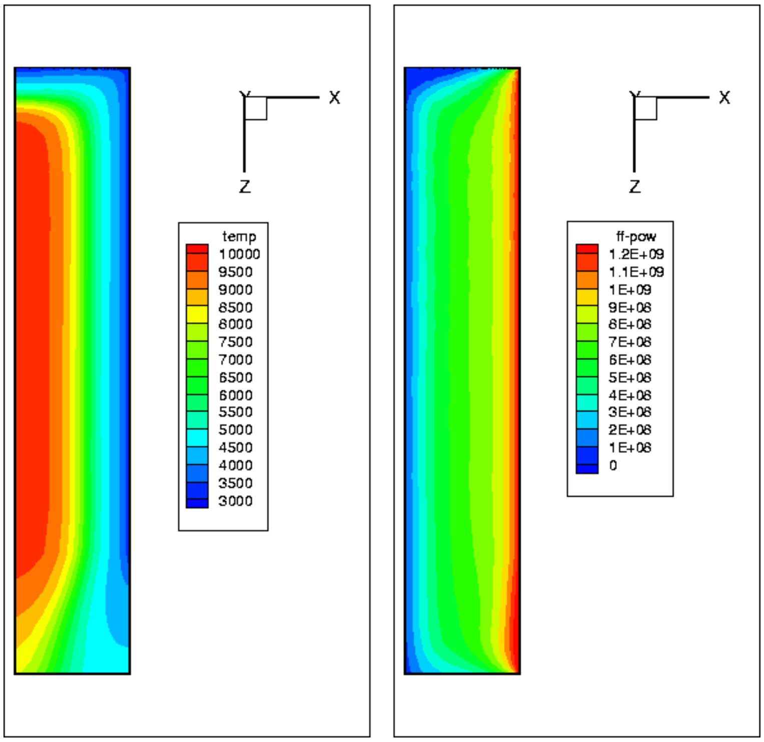

Figure 5: I risultati della simulazione.

risolti in passato con il pacchetto SUPER_ROCKET per poterne testare la validitá e la velocitá.

Il codice é stato quindi interfacciato direttamente con il solutore RANS (Reynolds Average Navier-Stokes) nativo del CRS4 denominato KARALIS. In maniera simile a quanto avveniva con il pacchetto SUPER_ROCKET il solutore RANS puó venire inizializzato con delle mappe di potenza sia costruite ad hoc sia provenienti da simulazioni precedenti. Si suppone quindi che il campo di densitá di potenza ceduto dai frammenti di fissione al gas sia comunque a grandi linee noto. A partire da questa configurazione il codice RANS evolve verso una situazione stazionaria (si porta cioé a convergenza). Solo a questo punto viene attivata la piattaforma **FEMME** nella sua configurazione atta al riscaldamento da frammenti di fissione. In una situazione congelata avviene la simulazione Montecarlo descritta in precedenza.

In figura 5 viene riportata una visualizzazione grafica del risultato della simulazione. Le condizioni a partire dalle quali é stata effettuata sono riportate nella Tabella 1 mentre la Tabella 2 riporta i risultati in uscita. Il caso test utilizzato riprende la linea di ricerca portata avanti nel progetto ASI 242 per le simulazioni del riscaldamento dell'idrogeno in camera di combustione, maggiori dettagli si possono trovare nella referenza [3]. In questo caso la simulazione montecarlo agisce a partire dai risultati delle simulazioni RANS nella geometria completa di un elemento

RESULTS	
No. of FF histories:	100000000
Total energy released by FF:	.827 E+10 MeV
Total energy released by FF:	3.77 $MWatt$
Total energy released from fuel layer:	.505 E+10 MeV
Total energy released from fuel layer:	2.30 $MWatt$
Total energy deposited in H2:	.421 E+10 MeV
Total energy deposited in H2:	1.91 $MWatt$
Total energy re-entering in fuel layer:	.85 E+09 MeV
Total energy re-entering in fuel layer:	.386 $MWatt$
Fractions refer to of fragments with positive direction (1/2)	
Fract.(%) of FF energy leaving the fuel layer	61.052
Fract.(%) of FF energy deposited in H2:	50.81
Fract.(%) of FF energy re-entering in fuel layer:	10.24
Fract.(%) of FF re-entering in fuel:	36.98
Average number of integration steps per FF:	39.0
Maximun local value in voxels in processors:	54.76 Mev
Average power density in gas:	.424 E+09 W/Kg
Average power density in gas:	424.15 W/mg

Table 2: I risultati in uscita.

cilindrico del motore (Figura 3). Considerando che l'Americio (il materiale fissile responsabile dell'emissione dei frammenti) non é distribuito sull'intera superficie interna dell'elemento e che il dominio RANS é suddiviso in otto diversi blocchi, nel caso della simulazione montecarlo il pacchetto **FEMME** legge in ingresso solo cinque degli otto blocchi totali, dominio che corriscponde alla effettiva camera di combustione al di sopra dell'ugello. Oltre al numero di domini effettivi da leggere il codice richiede in ingresso di fornire il numero di prcessori sui quali effettuare il conto ed il numero di *eventi* che si intendono lanciare. Il totale degli eventi viene quindi distribuito equamente su tutti i processori che singolarmente effettuano la simulazione per poi alla fine riunire statisticamente i dati e fornire le corrette distribuzioni di energia. Il caso in esame riguarda un cilindro di raggio 0.2 m ed altezza 2.5 m sulle cui pareti interne é distribuito un sottile strato di Americio 242 (.3 $10^{-5}m$) sufficiente a portare il materiale al di sopra della soglia di criticitá. Quella rappresentata in Figura 5 altro non é che la mappa della distribuzione di densitá di potenza sulle celle della griglia, in pratica l'energia ceduta dai frammenti di fissione all'idrogeno.

Tramite l'iterazione di diversi cicli RANS-montecarlo la fluidodinamica raggiunge uno stato stazionario, si raggiunge quindi la convergenza simultanea dei campi fluidodinamici sulle mappe di potenza fornite dalle simulazioni **FEMME**. In Figura 5 sono appunto mostrate le mappe di densitá di potenza nel caso in esame.

4 Conclusioni

In questo report é stata presentata una nuova piattaforma software, **FEMME**, capace di interagire direttamente con il codice RANS nativo del CRS4. La peculiaritá del nuovo oggetto é la sua flessibilitá: difatti la sua semplicitá permette di utilizzarlo per simulare situazioni fisiche differenti in contemporanea e parallelamente al codice fluidodinamico. É stata quindi presentata una prima applicazione di questo nuovo strumento su casi giá testati in precedenza, in particolare se ne é verificata l'efficacia nel caso della simulazione del riscaldamento del gas all'interno della camera di combustione di un elemento base del razzo a propulsione nucleare. La bontá dei risultati, in perfetto accordo con quelli precedenti, e soprattutto la sua rapiditá e la sua ampia duttilitá, fanno della piattaforma **FEMME** un ottimo candidato per il supporto ai codici RANS in simulazioni fluidodinamiche nuove e complesse.

5 Referenze

[1] *Fission Fragments Heating For Space Propulsion*, C. Rubbia.

[2] *The CFD code KARLIS3D*, M. Mulas et al. *CRS4 Tech. Rep. 00/87.*

[3] *Project 242: Fluid dynamic final report* M. Mulas *CRS4 Tech. Rep. 00/94.*

[4] *Project 242: Montecarlo codes for space propulsion*, A. Varone, P. Pili and E. Leonardi *CRS4 Tech. Rep. 2000.*

[5] *The Mapping Code : Interfacing the Fluid Dynamics solver KARALIS and Fission Fragments Montecarlo codes.* A. Varone *CRS4 Tech. Rep. 01/60.*