



POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

Implementation of a stochastic numerical method for the solution of the Euler equations for a supersonic flow

TESI DI LAUREA MAGISTRALE IN
AERONAUTICAL ENGINEERING - INGEGNERIA AERONAUTICA

Matilde Maria Cacopardi, 927663

Abstract: The aim of this work is to propose a variation of the DSMC method in order to be able to solve the Euler equations for a supersonic flow where a shock wave occurs due to the presence of an obstacle - or a solid wall more in general- inside the computational domain. The context of interest is to provide, after this study, a first step in order to simulate a flow surrounding a free rigid body starting from the computation of the forces considering the momentum transfer of the flow particles which collide with the body itself. The idea was born from the work [1] in which through the DSMC method the flow in the surroundings of a 1 mm meteoroid is analysed. In this way it would be possible to avoid to use a classical CFD software when the objective of the analysis is the evaluation of the global drag coefficient and a first idea of what happens in the flow field. The CFD method is powerful when a detailed description of the flow is needed, but it requires also a lot of effort to check the convergence of the problem, the mesh generation and the grid convergence even for a simple problem. For this reason, the idea is to provide a stochastic method that works directly with the single particles as the DSMC method does. As the computational cost of the DSMC method is proportional to the number of particles of the flow, a variation is needed in order to use it even for flows in standard condition in terms of temperature and pressure when the molecular mean free path is in the order of nanometers. The code is implemented in order to provide a solution simulating a fixed number of particles in each cell of the computational domain with a specific weight for each particles depending on the cell they belong to. The product of this work is just the main body of what is needed to be done in order to propose a competitive alternative to a CFD code. To follow the correct implementation of the code, three main simulations have been run by the stochastic code and by the CFD software SU2. The easiest simulation concerns an inviscid supersonic wedge; secondly a flat plate with a non-zero angle of attack and last the case of a rigid body with no degree of freedom inside the flow-field with an high Mach number.

Advisor:
Prof. Aldo Frezzotti

Academic year:
2021-2022

Key-words: DSMC, CFD, Shock wave

1. Introduzione

L'utilizzo di software di CFD per risolvere un campo di moto ad alto numero di Mach è una pratica diffusa ma, a seconda dall'obiettivo dell'analisi, il tempo richiesto per il set-up e il controllo della convergenza della simulazione, può risultare sproporzionato rispetto alla difficoltà dell'obiettivo di interesse: una grande quantità di tempo viene assorbita in primo luogo dal design della mesh che deve essere sufficientemente fine da poter considerare la soluzione indipendente dalla griglia; secondariamente anche la discretizzazione spaziale del flusso adeguata a catturare in modo sufficientemente accurato l'onda d'urto non è banale. Questo porta all'impiego di molto lavoro prima di poter avviare una simulazione che ci si aspetti andare a buon fine, anche nei casi di studio più semplici. Nel caso in cui si volesse prevedere con un'analisi preliminare l'entità dei carichi in gioco e una prima stima di dove si formerà l'urto e non ottenere un'analisi accurata di tutto il campo di moto, potrebbe convenire un approccio diverso. Questo può per esempio accadere sia se si tratta di verificare il set-up di prove sperimentali in galleria del vento sia se si tratta di una stima preliminare della forza di resistenza esercitata su un corpo tozzo di rientro dall'atmosfera. L'alternativa all'utilizzo di un software di CFD è il metodo DSMC [2]. Questo però, nel caso di simulazioni in condizioni standard ha comunque bisogno di un elevato numero di particelle perché si possano descrivere verosimilmente le proprietà macroscopiche del campo di moto; la dimensione della cella Δs è quindi un parametro fondamentale: questa deve essere piccola rispetto alla scala di lunghezza del gradiente delle quantità macroscopiche; la letteratura suggerisce in prima approssimazione $\Delta s = \frac{1}{3}\lambda$. Poiché λ è nell'ordine di grandezza del nanometro per un generico flusso in condizioni normali, il metodo diventa piuttosto impraticabile se l'obiettivo è sviluppare una strategia per ottenere certi risultati con il minor sforzo possibile. Infine si ricorda che, dal momento che un problema stazionario è simulabile solamente come stato asintotico del problema non-stazionario, sono necessari innumerevoli time-steps anche per il problema più semplice aggiungendo altra onerosità al metodo in termini di tempo. La potenza del DSMC però è il principio secondo cui ciascuna particella numerica rappresenta un numero molto elevato di particelle reali, definendo come input il parametro di peso $F_{num} = \frac{N_{reali}}{N_{sim}}$ con cui pesare tutte le particelle; F_{num} rappresenta il rapporto tra il numero di particelle reali del flusso indisturbato fratto il numero di quelle simulate nel metodo DSMC. L'interesse di studio in questa tesi sta nell'implementare un codice DSMC stocastico che basa il suo principale input non su F_{num} che resta uguale per ogni particella in qualsiasi punto del dominio per tutta la durata della simulazione, ma sul peso w_p . Il peso w_p è attribuito alla singola particella ed è legato alla cella cui essa appartiene nel singolo time-step: in base al numero di densità del flusso n_0 e al numero di particelle presente nella cella j , si ottiene $w_{p_t} = \frac{n_{j_t}}{N_{p_c}}$ dove n_{j_t} è il numero di densità della cella j allo step t . Con questo metodo l'obiettivo è di risolvere le equazioni di Eulero [5] in varie condizioni:

$$R(U) = \frac{\partial U}{\partial t} + \nabla \cdot \bar{F}^C - S = 0 \quad (1)$$

con

$$U = \{\rho, \rho \bar{u}, \rho E\}^T$$

$$F^C = \begin{pmatrix} \rho \bar{u} \\ \rho \bar{u} \times \bar{u} + \bar{I} p \\ \rho E \bar{u} + p \bar{u} \end{pmatrix}$$

Si vuole in particolare fornire un codice di base da sviluppare ulteriormente per descrivere con questo metodo il movimento di un corpo rigido immerso in un flusso affiancando le equazioni di Eulero con quelle di Newton: in questo modo sarebbe possibile calcolare le forze a partire dalla quantità di moto ceduta dalle particelle del flusso che urtano il corpo con il corpo stesso e conseguentemente l'effetto che la presenza del corpo rigido ha sul flusso circostante. Per questo motivo, le condizioni al contorno sulle pareti solide presenti nel dominio sono costituite dalla condizione di riflessione, unica consistente con l'equazione di Eulero. Il codice sviluppato parte dal lavoro precedente di Pullin che ha presentato in [9] un metodo diretto per il calcolo del campo di moto di un gas perfetto in equilibrio instazionario definendo il metodo dei flussi all'equilibrio e dal lavoro di Macrossan [7]. La sezione 2 descrive il metodo numerico e come è stato implementato il codice (in linguaggio Fortran 90) e i suoi passaggi fondamentali all'interno del loop temporale; la sezione 3 riporta le simulazioni che sono state testate con il codice descritto in 2 e in 4 sono discussi i risultati delle simulazioni computate. In particolare tutte le simulazioni ottenute con il codice stocastico, abbreviato con 'codice Stoc.', sono state confrontate con le soluzioni ottenute con il software *SU2* [3] e che vengono considerate le

soluzioni di riferimento; per questa classe di simulazioni, le varie mesh valutate per la convergenza di griglia sono state generate con il software Gmsh [4]. Le simulazioni oggetto di interesse vengono raggruppate in tre famiglie:

1. Flusso su rampa
2. Flusso su lastra piana
3. Flusso intorno ad uno ostacolo (corpo rigido)

Il flusso sulla rampa viene simulato con numero di Mach $M = 2$ e $M = 6$ (sezione 3.1); il flusso che investe la lastra piana con angolo di incidenza non nullo è simulato con $M = 6$ (sezione 3.2); il flusso intorno ad un ostacolo è quello di maggior interesse e viene simulato a $M = 0$ per valutare il raggiungimento dell'equilibrio delle forze e una volta verificato viene simulato con $M = 6$ e confrontato con SU2. Per questo ultimo caso (sezione 3.3) sono state fatte tre differenti simulazioni denominate 'Sim.1', 'Sim.2' e 'Sim.3' per analizzare meglio i risultati a valle e a monte dell'ostacolo, come descritto in seguito. I risultati vengono confrontati in termini di coefficiente di resistenza e numero di Mach. L'esito del lavoro è raccolto nella sezione 5.

da qui il tentativo del lavoro di sviluppare un codice DSMC che attribuisca un peso a ciascuna particella, in base al numero di particelle presente nella cella jc e il conseguente numero di densità per cella al momento della collisione. In questo modo lo step della collisione non ha bisogno di seguire la traiettoria della singola particella né stabilire la coppia delle particelle con la più alta probabilità di collisione reciproca. L'unica cosa che conta è memorizzare quali particelle sono nella cella jc per attribuire il giusto peso alla cella stessa e poi considerare la collisione tra due particelle con indice consecutivo considerando che siano presenti nuovamente N_{part_c} . La scelta di un time-step dt sufficientemente piccolo permette di disaccoppiare la fase di spostamento del flusso dalla fase di collisione inter-molecolare.

2. Il metodo numerico

Questa sezione descrive come il codice numerico è stato sviluppato seguendo la successione temporale. Dopo aver stabilito l'ampiezza dell'intervallo di tempo Δt , al tempo t_0 le particelle vengono inizializzate all'interno delle celle secondo il parametro di input N_{p_c} che stabilisce il numero di particelle per cella costante per tutta la durata della simulazione. Inoltre le particelle vengono iniettate dalle pareti aperte del dominio di calcolo dopo un opportuno campionamento delle stesse. Con un Δt sufficientemente piccolo si è in grado di valutare l'avvezione delle particelle in termini del loro avanzamento ed eventuale urto con le pareti solide; successivamente vengono valutate le collisioni intermolecolari e il ciclo ricomincia a $t_n = t_0 + n \cdot \Delta t$ con l'inizializzazione delle particelle in ogni cella e quelle iniettate. Nel paragrafo 2.1 si riporta la fase di avvezione cioè il moto delle particelle e la loro interazione con le pareti solide nel dominio con le condizioni al contorno adeguate; Infine la sezione 2.2 descrive l'inizializzazione delle particelle nel dominio e il campionamento di quelle in ingresso a seguito del calcolo delle quantità macroscopiche basate su ogni cella e le collisioni inter-molecolari. Per il calcolo dell'energia interna, il codice implementato tiene conto dei gradi di libertà rotazionali e vibrazionali delle molecole che costituiscono il flusso con $\theta_{vib} = 3300K$. Il metodo presentato si basa come anticipato in Sezione 1 sul metodo DSMC che simula direttamente la fisica del gas analizzato descrivendo le collisioni tra le particelle e le loro interazioni introducendo modelli direttamente fisici. Il metodo standard prevede che il numero di particelle simulate sia molto elevato affinché il flusso in ciascuna cella del dominio possano rappresentare la distribuzione delle particelle fisiche nel flusso reale e per questo è ampiamente utilizzato nelle simulazioni relativamente ai gas rarefatti. Il secondo parametro principale in un metodo DSMC è quello temporale Δt poiché la durata della simulazione rappresenta il tempo fisico con l'unica approssimazione che Δt sia sufficientemente piccolo da disaccoppiare il moto delle particelle e la loro collisione reciproca: il criterio con cui vengono selezionate le particelle che collidono quindi è strettamente legato al Δt stesso. Il primo criterio implementato da Bird [2] il metodo 'time-counter' ed è così strutturato:

1. scelta randomica di una coppia di particelle in una cella
2. valutando il rapporto tra e il massimo del valore raggiunto nella cella si applica il metodo 'acceptance-rejection' [6] per determinare se la coppia è selezionabile.
3. nel caso la coppia selezionata passi la valutazione del punto precedente, si associa un nuovo Δt alla cella chiamato Δt_c che dipende dal numero di densità e dal numero di particelle presenti nella cella ottenendo anche la frequenza media della collisione inter-molecolare e quindi il numero di collisioni nel tempo Δt

2.1. Avvezione e condizioni al contorno

Ad ogni time-step è necessario valutare se l'avanzamento delle particelle all'interno dello stesso time-step causa un urto con il corpo, con i bordi del dominio oppure se le particelle procedono indisturbate. Definendo t^* come il tempo che la particella jp ha a disposizione per l'impatto, inizialmente questo è pari a tutto l'intervallo di tempo Δt . Valutando la componente di velocità della particella normale a ciascuna superficie candidata all'urto con jp , ossia la velocità v_n , e la distanza da essa, si stima lo spazio percorso nel tempo di impatto t^* . Se la particella urta con la superficie S_i viene aggiornato il tempo di impatto e l'urto con minore tempo di impatto sarà l'unico urto verificatosi con la particella jp . Si considera infatti che ciascuna particella possa avere un solo urto all'interno del singolo time-step. A seconda delle condizioni al contorno implementate, la particella viene istantaneamente respinta e dunque si spostano con un moto rettilineo uniforme rappresentabile come un susseguirsi di linee spezzate. Per ogni urto viene immagazzinata la differenza di quantità di moto nelle due direzioni $x - y$ così da poter ottenere le forze al termine della durata dell'intera simulazione. La condizione al contorno delle superfici chiuse del dominio (o delle superfici dell'ostacolo) è implementata seguendo l'equazione di riflessione speculare:

$$\vec{v}_2 = \vec{v}_1 - 2(\vec{v}_1 \cdot \hat{n})\hat{n} \quad (2)$$

con \hat{n} la normale alla superficie di impatto, \vec{v}_1 il vettore di velocità prima dell'urto e \vec{v}_2 la velocità dopo l'urto. Questa è una semplificazione che elimina la dipendenza della riflessione dalla temperatura. Nel caso di superficie orizzontale, l'equazione precedente significa dunque che se una particella arriva contro la suddetta superficie con $\vec{v}_1 = (v_{x1}, v_{y1})$, verrà riflessa con $\vec{v}_2 = (v_{x1}, -v_{y1})$ e questa è l'unica condizione compatibile con l'equazione di Eulero. La fase di avvezione termina quando sono stati analizzati tutti t^* candidati perché ciascuna jp possa urtare una superficie. A questo punto il codice passa a rivalutare le nuove proprietà macroscopiche cella per cella. Le particelle infatti hanno cambiato la loro posizione e con il peso w_p daranno un contributo alla nuova cella di appartenenza. Si capisce subito come sia fondamentale per questo approccio aggiornare l'indice della cella cui jp appartiene mentre perde importanza e utilità seguire la particella nella sua particolare traiettoria. Nella cella jc non si troveranno più Np_c particelle ma quelle che sono arrivate dalle celle adiacenti in seguito all'avvezione considerando un numero totale di particelle pari a $Np_c \cdot Nc_{free} + Np_{in} - Np_{out}$ con

- Nc_{free} numero di celle non interne
- Np_{in} numero di particelle iniettate all'inizio del time-step
- Np_{out} numero di particelle che in seguito all'avvezione si trovano oltre le superfici aperte del dominio.

Ora la densità associata a ciascuna cella jc viene calcolata sommando tutti i w_p per p_{jc} . Dal calcolo delle velocità si sottrae l'energia cinetica della cella all'energia interna calcolata dalla somma pesata dell'energia vibrazionale più quella rotazionale di ciascuna particella e iterativamente si trova la temperatura necessaria per ottenere quell'energia interna.

2.2. Collisioni ed inizializzazione delle particelle

Con l'aggiornamento delle proprietà macroscopiche si considerano nuovamente Np_c particelle in ciascuna cella per valutare la collisione tra due particelle consecutive appartenenti a ciascuna cella. Il peso w_p e le velocità per ciascuna particella vengono dunque ri-aggiornate in base al numero di densità della cella e alla velocità macroscopica ad essa attribuita. Si assume che le due particelle che collidono avranno una velocità uguale in modulo scomponendo le tre componenti lungo la sfera di collisione nello spazio delle velocità [8]. Al termine di questa fase il time-loop riparte dall'iniezione di nuove particelle. Il flusso in ingresso attraverso la superficie aperta denominata 'superficie di inlet' è implementato secondo il metodo 'acceptance-rejection' denominato ' $A - R$ ' [6]. Considerando la funzione di distribuzione di Maxwell [8]:

$$f(v) = \frac{n}{(2\pi RT)^{3/2}} \exp - \frac{(\mathbf{v} - u\hat{\mathbf{x}})^2}{2RT} \quad (3)$$

si è in grado di descrivere la funzione di distribuzione di probabilità pdf nelle tre variabili random v_x , v_y e v_z a partire dalla densità del numero di flusso attraverso una generica superficie definita dalla normale $\hat{\mathbf{x}}$. Le tre variabili rappresentano la velocità della particella nel suo complesso, ossia la sua

velocità media e quella peculiare.

$$v_x f(v) = nv_x \frac{\exp -\frac{(v_x-u)^2}{2RT}}{(2\pi RT)^{1/2}} \times \frac{\exp -\frac{v_y^2}{2RT}}{(2\pi RT)^{1/2}} \times \frac{\exp -\frac{v_z^2}{2RT}}{(2\pi RT)^{1/2}} \quad (4)$$

La più probabile velocità assunta da una particella entrante, dipende dalla velocità media del flusso in quella direzione, dal gas e dalla temperatura e derivando l'equazione precedente si ottiene che $v_{max} = \frac{\sqrt{(u^2+4)+u}}{2}$. Si ricava dunque il flusso ad essa associato:

$$g_{max} = v_{max} \frac{\exp -\frac{(v_x-u)^2}{2RT}}{(2\pi RT)^{1/2}} \quad (5)$$

Il metodo $A-R$ entra in gioco qui: generando una velocità v_x in un intervallo abbastanza ampio nello spazio delle velocità così da contenere v_{max} , se il flusso ad essa attribuito si trova all'interno della campana propria della funzione di distribuzione, allora la particella con quella velocità viene accettata (fig. 1). Il processo di inlet è quindi descritto come un processo in cui il verificarsi del singolo evento è indipendente dal verificarsi di quello precedente, trascurando la probabilità che più eventi possano verificarsi nello stesso momento. Il processo è quindi descritto come un processo di Poisson e un adatto campionamento degli eventi risulta fondamentale: ogni flusso entrante contribuisce a definire la porzione di time-step utile per l'ingresso di una nuova particella nel dominio. Le variabili random v_y e v_z sono generate invece dalla distribuzione di probabilità maxwelliana.

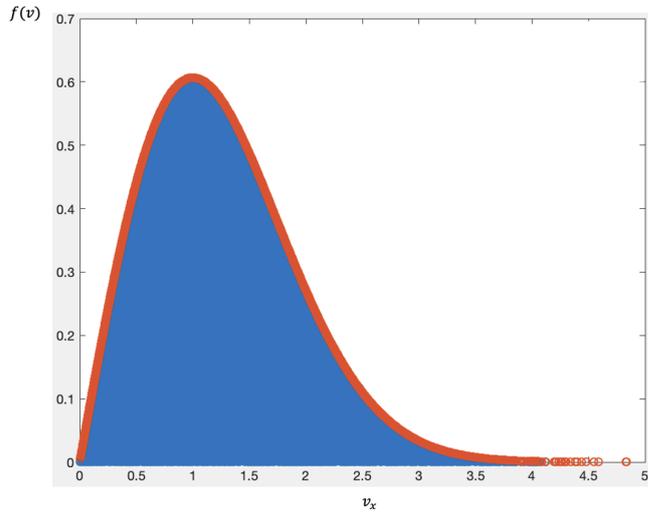


Figura 1: Funzione di distribuzione di v_x in rosso. L'area sottesa contiene i valori accettati

Concludendo, le particelle p_{in} iniettate entrano nel dominio ciascuna al tempo $t_n + \delta\tau_p$ e si aggiungono a quelle che vengono implementate già all'interno del dominio. Inoltre, le particelle iniettate vengono caratterizzate dalla posizione casuale lungo $x = x_{in_p}$ dove x_{in_p} è la posizione p_0 della particella p da cui viene calcolata l'effettiva posizione all'interno del dominio; questa dipende dalla velocità attribuita loro e dal $\delta\tau_p$. La strategia è un espediente per non trovarsi nella situazione iniziale di N particelle disposte tutte in fila lungo la medesima x_{in} che procedono in parallelo. All'inizio di ogni time-step vengono inizializzate le particelle secondo il parametro fissato Np_c , che come detto, stabilisce il numero di particelle presenti in ciascuna cella o porzione di cella; ad esse viene attribuita una posizione casuale e la velocità peculiare secondo la distribuzione di probabilità Gaussiana così come per l'energia rotazionale e vibrazionale

$$v_i = \sqrt{-2 \log(1 - rf()) R_{gas} T} \quad (6)$$

In questa fase, poiché il riempimento del dominio avviene cella per cella, è fondamentale stabilire la porzione di cella libera disponibile ad accogliere le particelle. Le celle infatti, a seconda della geometria del problema in analisi, possono essere o totalmente libere per le particelle o totalmente occupate dalla presenza di un ostacolo nel dominio oppure ibride, cioè occupate dall'eventuale corpo solo in parte.

Perché questa operazione si possa compiere agevolmente, il dominio implementato nel codice Stoc. è un dominio quadrato suddiviso in $N_{c_x} \times N_{c_y}$ celle, ciascuna di dimensioni $dx \times dy$. Schematizzando l'eventuale ostacolo come un corpo rigido quadrato, questo viene riempito da particelle fittizie con una area pari al 10% dell'area della cella. Esse vengono posizionate una adiacente all'altra riempiendo per righe tutto il corpo rigido. Le particelle fittizie vengono caratterizzate dalla loro posizione e dall'indice della cella cui appartengono: per stimare la porzione di cella disponibile per le particelle reali, basta sottrarre l'area della particella fittizia ogni volta che se ne trova una nella cella jc . Indicando con A l'area della cella adimensionalizzata su $dx \cdot dy$, si ottengono tre situazioni:

1. $A_j = 1$: la cella j è da riempire di particelle di flusso in tutta la sua area $dx \cdot dy$, cioè j è una cella esterna
2. $A_j = 0$: la cella j è completamente occupata dall'ostacolo e dunque non verrà riempita dalle particelle di flusso, cioè è una cella interna
3. $0 < A_j < 1$: la cella j verrà riempita dalle particelle di flusso solo per la frazione di cella $(1 - A_j) \cdot dx \cdot dy$, la cella è mista.

L'inizializzazione delle particelle viene fatta in primo luogo riempiendo facilmente le celle vuote in posizione $pos = pos_{C_{ij}} + rf() \bar{s}$, dove $pos_{C_{ij}}$ è la coordinata della cella i, j e $rf()$ è la funzione che genera numeri randomici nell'intervallo aperto $(0, 1)$. Infine si passa alle celle miste e in questo caso s sarà solo la porzione libera della cella. Inoltre le particelle sono inizializzate con la loro energia vibrazionale e rotazionale e_v, e_r e con il loro peso w . Come detto, quest'ultimo è il parametro fondamentale per il codice stocastico dal momento che è proprio l'espedito che permette di descrivere il flusso con un numero molto inferiore di particelle; il peso dipende dal valore di Np_c e indica il numero di particelle reale che la singola particella numerica sta rappresentando:

$$w = \frac{n_0}{Np_c} \quad (7)$$

dove n_0 è il numero di densità delle particelle $[part]/[m^3]$. Al tempo t_0 , $w = w_{inf}$ costante per tutte le celle. Una volta definite le proprietà microscopiche vengono definite quelle macroscopiche del flusso cella per cella, ossia le variabili che caratterizzavano le particelle numeriche vengono tradotte in variabili che descrivono il flusso fisico: per ogni cella jc si ottiene il numero di densità come somma dei pesi di tutte le particelle che si collocano nella cella in questione; la velocità numerica diventa fisica considerando $u_i = w \cdot v_i$ e la velocità macroscopica è calcolata come somma vettoriale di tutte quelle fisiche all'interno della cella. Terminata la fase di inizializzazione e di determinazione delle proprietà macroscopiche cella per cella a $t = t_0$, si passa a seguire nuovamente il moto delle particelle e la loro interazione (par.2.1).

3. Simulazioni

In questa sezione vengono descritte le simulazioni testate con il codice Stoc. e con il software SU2 limitatamente al set-up mentre i risultati saranno riportati nella sezione successiva (sezione 4) come anticipato nella sezione 1. Il flusso analizzato a diversi numeri di Mach, è un flusso di azoto N_2 .

3.1. Rampa

Il problema viene simulato per testare il comportamento del codice Stoc. in presenza della formazione di un'onda d'urto obliqua nel campo di moto del fluido inviscido: l'interesse è valutare con che errore si è in grado di stimare la posizione dello shock rispetto ai dati di riferimento di SU2 e confrontare l'errore ottenuto sul coefficiente di resistenza. I dati di riferimento sono resi disponibili da una simulazione prodotta con il software SU2 che risolve per via numerica le equazioni del moto descritte dalle equazioni di Eulero comprimibile. Il caso è particolarmente semplice in quanto è possibile risolverlo anche con le relazioni di salto di Rankine-Hugoniot [10]. Il flusso viene simulato a due diversi numeri di Mach a parità di geometria del problema. Il dominio rettangolare è delimitato da una superficie di ingresso e uscita (parete sinistra e destra rispettivamente), da una parete solida superiore e dalla parete inferiore che si sviluppa in una rampa con angolo di 10deg (fig. 2). Il flusso entra con velocità orizzontale. La simulazione di SU2 è stata risolta su quattro differenti griglie per ottenere la risoluzione che garantisce il più alto livello di non-dipendenza della soluzione dalla mesh. In tutti i casi la mesh è una griglia strutturata che divide la geometria in due parti: la prima parte è costituita dal blocco di sinistra che si estende dalla superficie di ingresso fino alla rampa ed è composto dal susseguirsi di quadrati nelle

due direzioni $x - y$; la seconda parte è costituita dal blocco di destra delimitato inferiormente dalla rampa stessa. Gli elementi di griglia in entrambi i blocchi hanno i quattro lati paralleli ai quattro segmenti che delimitano il blocco stesso così da seguire al meglio la direzione del flusso medio. Per

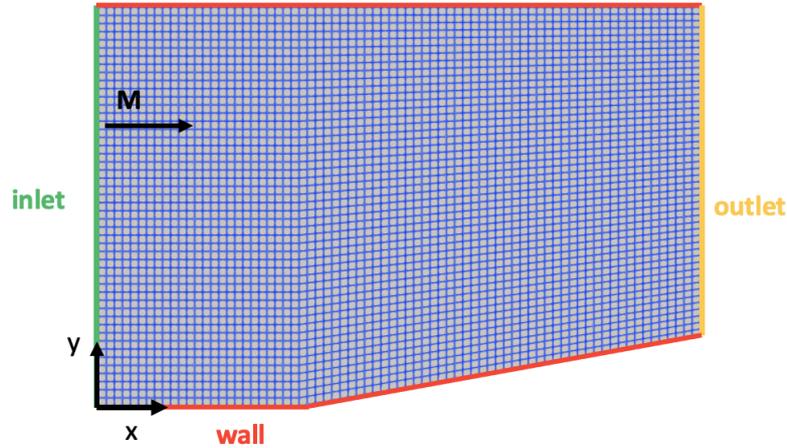


Figura 2: Mesh per la rampa in SU2 e condizioni al contorno

entrambi i flussi a $M = 2$ e $M = 6$, le equazioni di Eulero del problema $2D$ vengono discretizzate nello spazio con lo schema convettivo HLLC del secondo ordine: lo schema è di tipo TVD, 'up-stream' ed è un metodo 'shock-capturing' costruito a partire dal più generale metodo di Gudonov che approssima la soluzione con la costruzione del problema di Riemann ad ogni interfaccia di due celle consecutive; la discretizzazione temporale per ciascuna delle iterazioni del ciclo interno è compiuta con il metodo Eulero-Implicito [5]. La simulazione arriva a convergenza con il controllo dei residui sui valori della densità e il numero di Courant $CFL = 2$. La prima mesh è costituita da 75×50 elementi per un totale di 3750 nodi ed è la più lasca: le altre tre sono costruite con un numero di elementi via via crescente come riportato in tabella 1 per entrambe le simulazioni. Sulla parete inferiore e quella superiore viene imposta la condizione di non penetrazione senza slittamento; la condizione della superficie di ingresso è costituita dal marker 'supersonic inlet' disponibile su SU2 e si specificano le condizioni del flusso indisturbato, in particolare la sua temperatura, la pressione statica e il vettore velocità; la condizione per la superficie di uscita è specificata invece dal Marker 'supersonic outlet'. Di seguito si riportano le proprietà macroscopiche del flusso indisturbato:

- $M_{1\infty} = 2$; $M_{2\infty} = 6$
- $T_{\infty} = 300K$
- $p_{\infty} = 100000Pa$
- $\vec{V} = (695.43, 0, 0)^T$

	Numero di elementi	
	$M = 2$	$M = 6$
$m1$	75×50	75×50
$m2$	90×50	90×90
$m3$	90×90	120×90
$m4$	120×100	150×100

Tabella 1: Mesh generate per la simulazione della rampa

Il flusso entra con angolo d'attacco nullo. La stessa geometria è stata implementata sul codice fortran ma è stato necessario adattare la routine che si occupa del riempimento delle particelle nel dominio all'inizio di ogni time-step poiché il meccanismo della procedura di riempimento delle celle si basa sulla determinazione della frazione di cella disponibile. Per semplicità il dominio è costruito come un rettangolo $(x_{min}, x_{max}) \times (y_{min}, y_{max})$ e le celle vengono raggruppate secondo il criterio già illustrato nella sezione 2.2:

- celle esterne: quelle completamente disponibili al riempimento. Queste si troveranno in tutto il blocco del dominio prima della rampa e immediatamente sopra ad essa
- celle interne: quelle che non verranno riempite dal flusso. Queste si troveranno nella porzione di dominio al di sotto della rampa
- celle miste: quelle con una frazione utile al riempimento inferiore ad 1 rispetto alla loro area. Queste si troveranno lungo la rampa.

Si illustrano di seguito le condizioni analitiche con cui ciascuna cella jc viene attribuita a uno dei tre gruppi:

1. la rampa viene costruita tramite la retta $Y = \tan \frac{\pi}{18} \cdot (x - x_1)$ con $x_1 = 0.5$ la coordinata orizzontale da cui si sviluppa la rampa ($y_1 = 0$)
2. se l'estremo destro della cella è inferiore o uguale a x_1 la cella è sicuramente esterna
3. se l'estremo sinistro della cella è superiore o uguale a x_1 la cella può essere esterna, interna o mista:
 - (a) si valuta la posizione y della rampa in corrispondenza dell'estremo destro della cella e se questa è inferiore o uguale all'estremo inferiore della cella stessa allora anche in questo caso la cella sarà esterna
 - (b) se la condizione precedente non è verificata e l'estremo superiore della cella è maggiore della y di intersezione cella-rampa, allora sarà una cella mista.
4. calcolando le posizioni y_i della rampa in corrispondenza degli estremi sinistri e destri della cella, questa sarà interna se il suo estremo superiore si mantiene inferiore o uguale a y_i .

La cella interna non viene riempita, la cella esterna viene riempita generando delle posizioni casuali all'interno di dx e dy per le due componenti; la cella mista viene riempita generando casualmente le posizioni lungo la direzione x nello spazio delimitato dal suo estremo inferiore e dalla coordinata x_i che esprime l'intersezione con la rampa. Per ciascuna posizione così generata si trova la corrispondente intersezione y_i con la rampa e la posizione verticale viene casualmente assegnata in uno spazio compreso tra y_i e l'estremo superiore della cella analizzata. Si rimanda allo schema di immagine 3. Una primo test è stato fatto con la risoluzione data da 75×50 celle con 50 particelle per cella e poi un secondo test raddoppiando la risoluzione. Le quantità macroscopiche di interesse sono calcolate su tutte le celle non interne ogni 100 time-steps e poi raggruppati a partire dallo step in cui si raggiunge l'equilibrio del numero di particelle presenti in seguito al loro spostamento nel dominio o fuori da esso. L'ampiezza dello step temporale è omogenea e pari a 10^{-5} s per un totale di 100000 step. L'ampiezza del passo temporale è stato scelto così da restare inferiore alla metà del tempo impiegato dal flusso indisturbato per attraversare una singola cella. Per ottenere il coefficiente di resistenza si immagazzina la somma della differenza di quantità di moto di ciascuna particella impattante con la rampa pesata per il peso che le compete all'interno di ogni time-step:

$$\Delta q \bar{d} m = \Delta q \bar{d} m + w_p m \cdot (\bar{u}_2 - \bar{u}_1) \quad (8)$$

3.2. Lastra piana

La simulazione del problema prevede una lastra piana nel centro di simmetria del dominio e si vuole analizzare la bontà del coefficiente di resistenza per un flusso che la investe a Mach $M = 6$. Sono state performato due geometrie con il codice stocastico e con il risolutore di riferimento SU2. Nel primo caso il problema è valutato in un dominio di calcolo quadrato con le seguenti coordinate:

- $(x_{min}, x_{max}) = (-0.001, 0.001)$ m
- $(y_{min}, y_{max}) = (-0.001, 0.001)$ m

La lastra piana passa per il centro con un angolo di attacco pari a $\alpha = 26.5$ deg rispetto all'orizzontale e coordinate

- $(x_1, y_1) = (-0.0002, -0.0001)$ m
- $(x_2, y_2) = (0.0002, 0.0001)$ m

La figura 4 riporta la geometria del problema implementata nel codice stocastico. L'immagine riporta un numero più lasco di elementi di griglia per maggiore chiarezza. La routine di riempimento delle particelle nel codice stocastico è stata nuovamente adattata alla geometria del problema analogamente con quanto fatto con il caso della rampa in sezione 3.1. In questo caso però la programmazione risulta più agile in quanto non c'è la necessità di raggruppare le celle in interne, esterne e miste: si valutano le intersezioni della retta che descrive la lastra piana con i quattro estremi della cella: se non ci sono intersezioni le posizioni randomiche delle particelle vengono assegnate considerando l'intera area

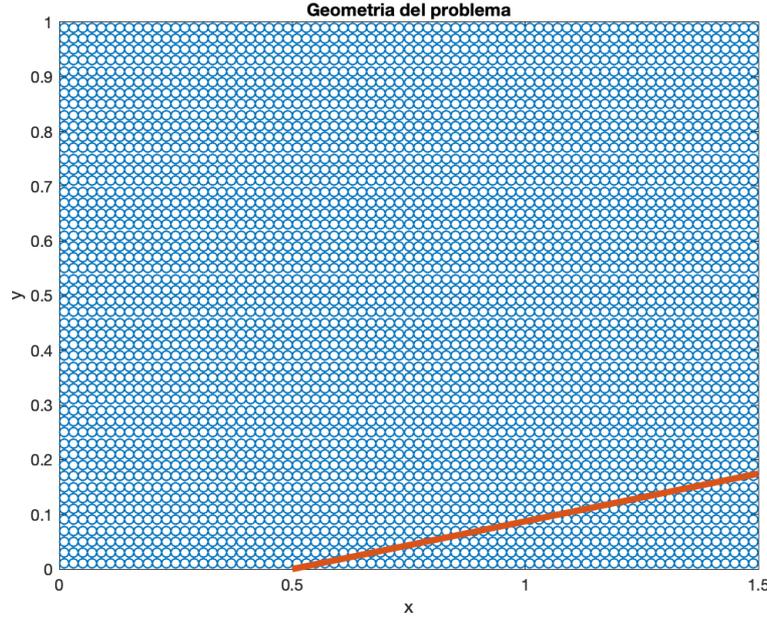


Figura 3: Dominio di calcolo per il codice Stoc.

$dx \times dy$. Se si registrano intersezioni, si generano randomicamente le posizioni y_p ; in questo caso la posizione x_p viene assegnata a sinistra o a destra della lastra a seconda che l'indice della particella sia pari o dispari. In questo modo si garantisce uniformità delle particelle nella singola cella. Sono simulate 100 particelle per cella con la risoluzione ottenuta dalla griglia composta da 75×50 celle. Lo spazio temporale è stabilito pari al tempo che la particella impiega a compiere un quarto della cella nella direzione x con velocità media u_m . La mesh per SU2 è anch'essa una mesh strutturata formata da 50×30 elementi. Per semplicità, si sfrutta in questo caso la simmetria del problema e si impone un angolo di incidenza alla corrente $\alpha = -26.5\text{deg}$ per mantenere la lastra con un angolo d'attacco nullo. Sfruttando la simmetria si ottiene per SU2 un dominio di calcolo suddiviso in tre blocchi:

- il blocco a sinistra prima della lastra
- il blocco centrale delimitato inferiormente dalla lastra stessa
- il blocco di destra che segue la lastra

Le condizioni al contorno che vengono imposte sono quelle di simmetria del flusso per il bordo inferiore dei due blocchi alle estremità e le condizioni di non slittamento e non penetrazione sulla lastra. La corrente entra dalla superficie laterale sinistra e da quelle superiori mentre esce da quella laterale destra. Le condizioni del flusso indisturbato sono riportate di seguito:

- $p_\infty = 100000\text{Pa}$
- $T_\infty = 300\text{K}$
- $M_\infty = 6$

Nella simulazione di SU2, nella condizione di 'supersonic-inlet' viene specificato il vettore di velocità esplicitando le componenti di velocità nelle direzioni $x - y$ mentre nel codice stocastico, in cui non si sfrutta la simmetria, la velocità indisturbata è puramente orizzontale. Le dimensioni geometriche del primo test portano entrambi i codici numerici a divergere: per quanto riguarda SU2 la divergenza si ottiene con diversi schemi numerici del secondo ordine e l'impostazione di un basso numero di Courant riesce a produrre dei risultati per niente rappresentativi solo con un metodo al primo ordine anche con una mesh molto fitta. La situazione migliora aumentando le dimensioni del dominio in direzione y . Le proprietà della corrente indisturbata non vengono variate, ma viene aumentata la scala di lunghezza del problema per ottenere dei risultati adatti:

- $x_{min} = -1\text{m}$
- $x_{max} = 1\text{m}$
- $y_{min} = -1\text{m}$
- $y_{max} = 1\text{m}$
- $(x_1, y_1) = (-0.4, -0.4)\text{m}$

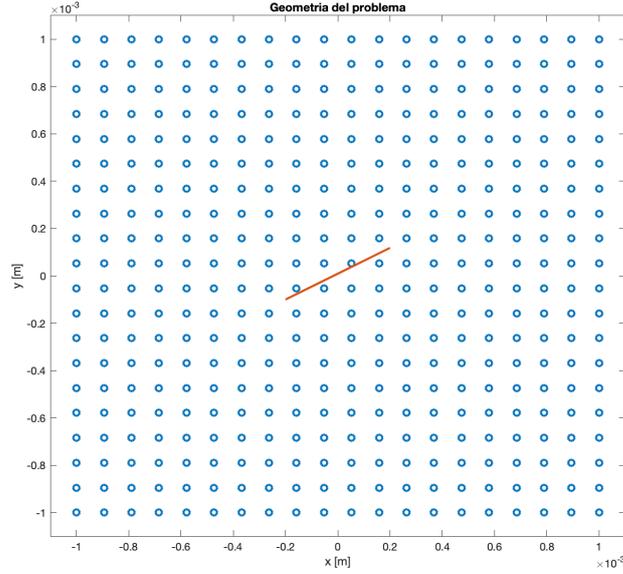


Figura 4: Dominio di calcolo per la prima simulazione

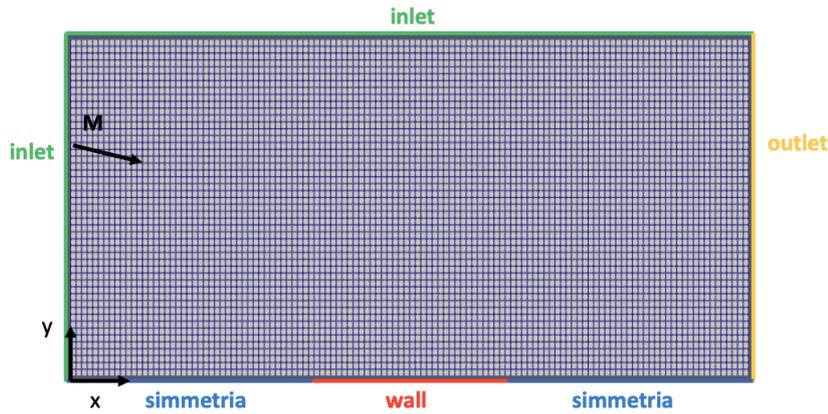


Figura 5: Mesh per la simulazione di SU2

- $(x_2, y_2) = (0.4, 0.4)\text{m}$

La soluzione di riferimento viene ottenuta in questo caso con uno schema convettivo *TVD* del tipo centrale *JST* e numero di Courant $CFL = 1$. Sono state generate quattro mesh denominate $m1$, $m2$, $m3$, $m4$ con un numero di elementi via via crescente che si possono osservare in tabella 2. Anche la risoluzione del dominio implementato nel codice stocastico viene aumentata come viene riportato nella sezione dedicata ai risultati 4.2.

3.3. Flusso intorno ad un ostacolo

3.3.1 Mach = 0

Questo problema è simulato per raggiungere una prima validazione del codice stocastico in termini di raggiungimento dell'equilibrio sia per quanto riguarda il numero di particelle in ingresso ad ogni time-step sia per quanto riguarda le forze esercitate dal corpo sul flusso. Si simula dunque un flusso di azoto N_2 a temperatura e pressione ambiente in un dominio quadrato che investe un corpo senza gradi di libertà posto nel centro come rappresentato in figura 6. Di seguito le coordinate che descrivono la geometria del problema:

- Dominio: $P_1 = (-0.05, -0.05)$; $P_2 = (0.05, -0.05)$; $P_3 = (0.05, 0.05)$; $P_4 = (-0.05, 0.05)$.

	Numero di elementi
$m1$	170×50
$m2$	200×80
$m3$	300×100
$m4$	400×150

Tabella 2: Numero di elementi per le quattro griglie

- Ostacolo: $(-0.025, 0.025) \times (-0.025, 0.025)$

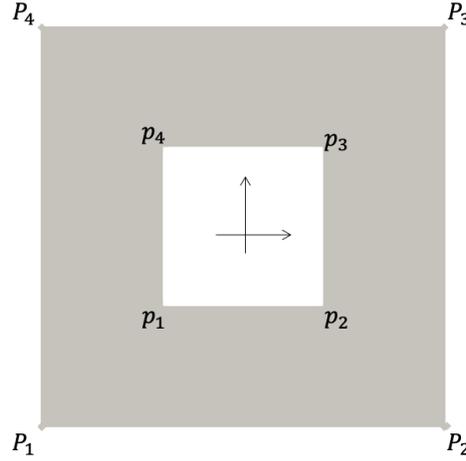


Figura 6: Geometria del problema simulato. Il dominio in grigio è interamente investito dal flusso, il corpo rigido in bianco è fermo al centro.

Come primo tentativo si è utilizzata una griglia formata da 40×20 celle nelle direzioni $x - y$; la risoluzione è senz'altro bassa ma ci si aspetta che nel caso presente in cui l'obiettivo è valutare la forza e quindi la differenza di quantità di moto delle singole particelle e non le proprietà del campo di moto, la bassa risoluzione non crei disturbi nel risultato. Le condizioni del flusso indisturbato sono le seguenti:

- Temperatura: $T_p = 293.15K$
- Pressione: $p_p = 101325Pa$
- Mach: $M = 0$.

Per le pareti superiori e inferiori del dominio e le superfici del corpo rigido all'interno, è stata implementata la condizione di riflessione speculare nel caso di urto con una particella. La superficie di destra e di sinistra del dominio sono aperte e in accordo con il metodo "acceptance-rejection" (2.2), le particelle con una certa velocità peculiare sono libere di entrare (o uscire); in particolare vengono iniettate all'inizio di ogni time-step e fatte uscire alla fine della fase di avvezione che valuta lo spostamento delle particelle e l'eventuale scontro con una delle superfici solide prima della fase di collisione. A parità di griglia sono state fatte diverse simulazione ciascuna con un diverso numero di particelle per cella, da un minimo di 18 ad un massimo di 100 così da avere anche una stima dei tempi di calcolo.

3.3.2 Mach = 6

Questa simulazione è stata progettata con lo scopo di testare il codice stocastico in un caso di flusso intorno ad un corpo tozzo e valutare la posizione dell'onda d'urto e del coefficiente di resistenza. La soluzione ottenuta verrà confrontata con la soluzione fornita con l'approccio numerico deterministico di SU2. Come illustrato in seguito, questo test si suddivide in tre simulazioni distinte indicate con Sim. 1, Sim. 2 e Sim. 3. In generale il dominio implementato segue lo schema di quello illustrato nella simulazione precedente. In particolare il corpo, di dimensioni molto inferiori rispetto al dominio, viene posto vicino alla superficie di uscita del flusso per concentrare l'analisi del campo di moto davanti al corpo stesso (Sim. 1). La geometria del problema si può osservare in figura 7: il dominio è costituito da 160×60 celle e ciascun lato del corpo rigido corrisponde a 2.65 e 3 volte il lato della cella in direzione

x e y rispettivamente (figura 8). Ovviamente una migliore risoluzione sarebbe preferibile ma la scelta

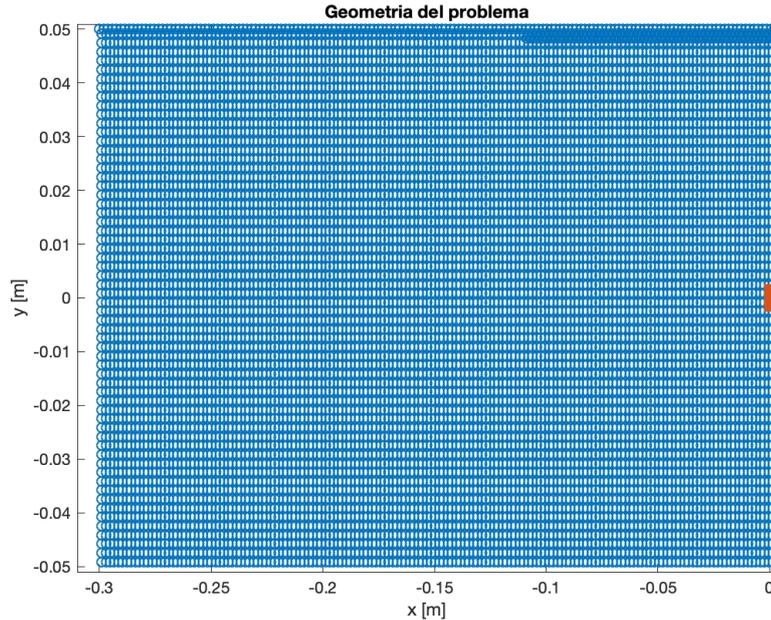


Figura 7: Geometria del problema implementato

si giustifica con la decisione di mantenere il tempo di calcolo inferiore a 5ore. A ogni time-step vengono implementate 5000 particelle per cella e iniettate dalle due superfici aperte a sinistra e destra. Come in precedenza, sulle superfici inferiori e superiori del dominio e su tutte le superfici del corpo rigido è imposta la condizione di riflessione speculare. La discretizzazione temporale viene fatta con un intervallo di tempo pari all' 1% del tempo impiegato dal flusso indisturbato per percorrere metà della cella in direzione orizzontale.

Per quanto riguarda la geometria processata da SU2, essa sfrutta la simmetria del problema simulando solo la metà superiore del campo di moto con le opportune condizioni di simmetria sul flusso lungo il bordo inferiore che precede il corpo rigido. Sono state sufficienti tre mesh per testare la convergenza di griglia, i cui parametri si possono vedere in tabella 3. Le mesh sono di tipo strutturato con un numero di elementi via via crescente. Le linee che costituiscono il corpo rigido e la superficie superiore del dominio presentano la condizione di non slittamento e non penetrazione mentre le due superfici verticali a sinistra e destra costituiscono le superfici di ingresso e uscita del flusso. Una seconda simulazione di questo tipo (Sim. 2) è stata fatta con un dominio più ampio per cercare di valutare anche gli effetti dell'onda d'urto a valle del corpo rigido. In questo caso sono state aumentate le dimensioni del corpo rigido di un fattore 10 e la parete di destra è stata traslata in $x_{max} = 0.3$ portando a 180 il numero di celle nella direzione x per quanto riguarda la simulazione stocastica. Per indagare l'eventuale riflessione dell'onda d'urto a valle del corpo, x_{max} è stata nuovamente spostata ad un valore $x_{max} = 10 \cdot y_{max}$, cioè $x_{max} = 5m$ (Sim. 3). Questo comporta l'ulteriore aumento del numero di celle in direzione x per il codice stocastico di $N_{c_x} = 300$. Di nuovo la risoluzione è piuttosto bassa ma è un compromesso che è stato accettato rispetto ai tempi di calcolo. Le tre griglie utilizzate nelle tre simulazioni stocastiche sono riportate in tabella 4.

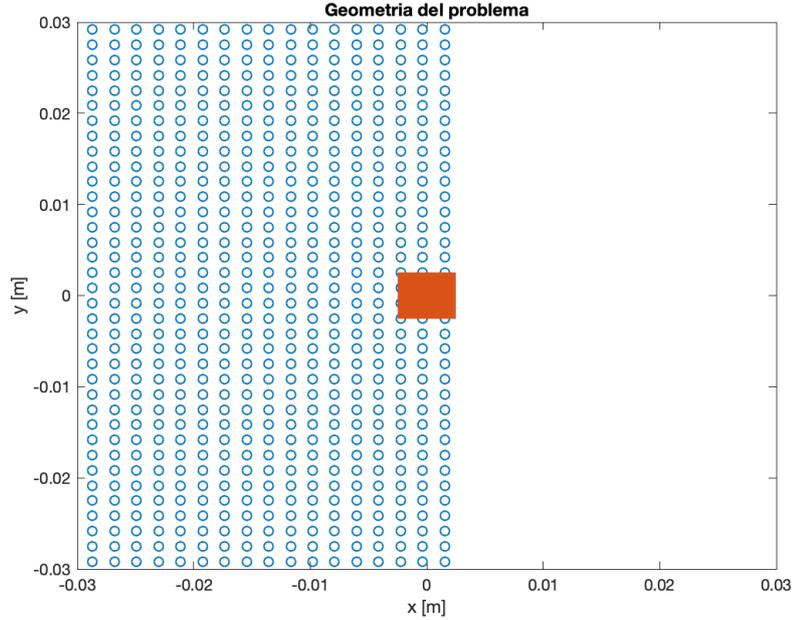


Figura 8: Geometria del problema - zoom sul corpo rigido

4. Risultati

In questa sezione vengono discussi i risultati ottenuti per le tre famiglie di simulazioni elencate nella Sezione 1.

4.1. Rampa

4.1.1 Mach = 2

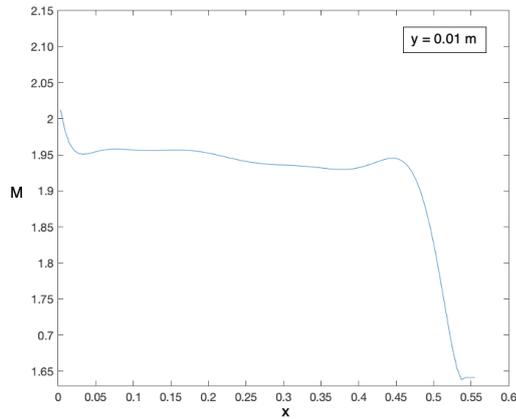
La soluzione di riferimento viene presentata con la mesh strutturata $m2$ (tab. 1) che presenta una risoluzione di 1×1 [mm] nel blocco prima della rampa con 750×100 celle. Si osserva in figura 10c l'andamento della densità lungo $y = 0.5$ m e del numero di Mach (fig.10b) che passa da 2 a 1.59; si può determinare la posizione dell'onda d'urto in $x_s = 1.086$ m. Il punto di formazione dell'onda d'urto in $x_s = 0.459$ [m] si osserva invece nel grafico di figura 10a. Il coefficiente di resistenza, oggetto di interesse, è pari a 0.044483 come riportato in tabella 5. Per quanto riguarda la soluzione ottenuta con il codice Stoc. implementato, si riportano il numero di Mach lungo $y = 0.01$ [m] e $y = 0.5$ [m] nei grafici di figura 9a e 9b rispettivamente. La risoluzione del campo di moto è pari a 0.5×1 [mm]. Sebbene l'andamento del numero di Mach abbia un comportamento per lo più oscillante prima di incontrare la rampa, la posizione dell'onda d'urto viene ben catturata ed è in accordo con quanto riportato dalla soluzione di riferimento. L'interesse sta nel fatto che si è comunque in grado di ottenere il coefficiente di resistenza con una tolleranza dello 0.0067% rispetto ad SU2 (tab.5): $C_D = 0.044480$

Numero di elementi	
Sim. 1	
$m1$	10000
$m2$	2500
$m3$	3500
Sim. 2	
$m1$	12000
$m2$	23500
$m3$	35000
Sim. 3	
$m1$	15000
$m2$	33731
$m3$	67462

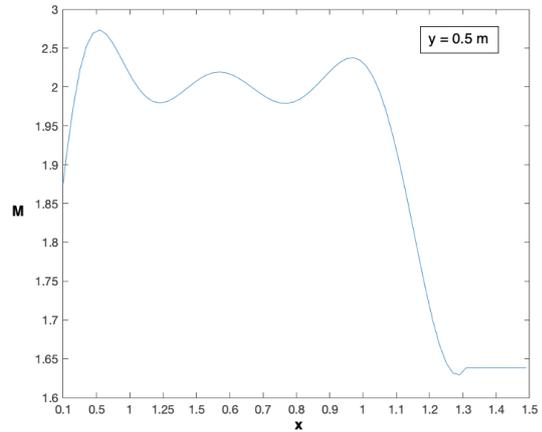
Tabella 3: Mesh generate per SU2 nelle tre simulazioni

Numero di celle	
Sim. 1	160×60
Sim. 2	160×80
Sim. 3	300×80

Tabella 4: Griglia per le tre simulazioni calcolate con il metodo Stoc.



(a) Rampa simulata con codice Stoc. Lo shock lungo $y = 0.01m$ si forma a x_s inferiore del 1.13% rispetto al caso simulato con SU2

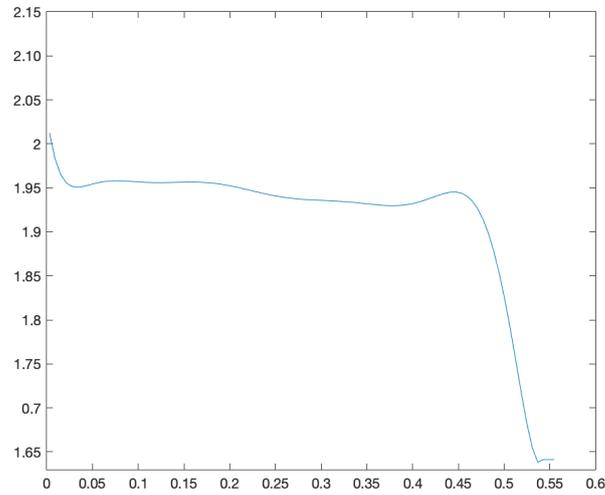


(b) Rampa simulata con codice Stoc. Oscillazione del numero di Mach prima della rampa in $y = 0.5m$

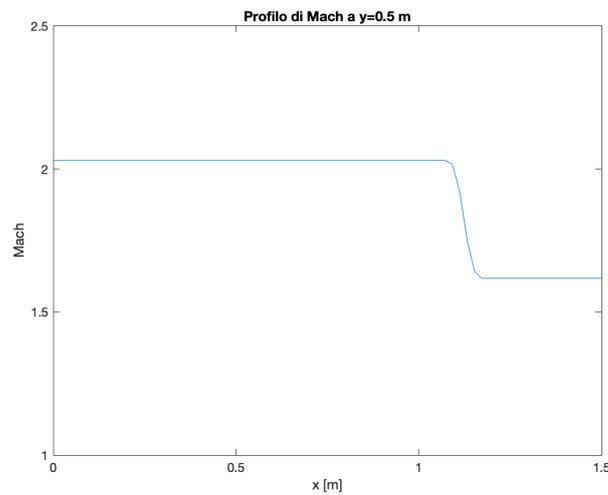
Figura 9: Rampa, risultati ottenuti con il codice Stoc., Mach=2

C_D	Codice Stocastico	SU2
	0.044480	0.044483

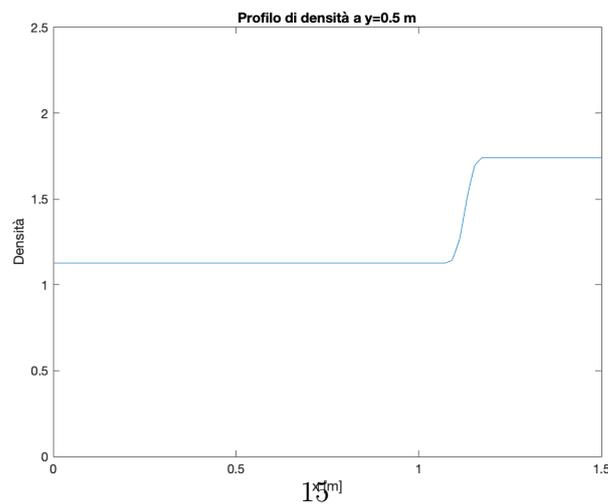
Tabella 5: Coefficiente di Resistenza ottenuto dai due codici



(a) Rampa simulata con SU2. L'onda d'urto lungo $y = 0.01m$ si forma in $x_s = 0.459$



(b) Rampa simulata con SU2. Numero di Mach prima e dopo l'urto lungo $y = 0.5m$



(c) Rampa simulata con SU2. Distribuzione della densità, $y = 0.5m$

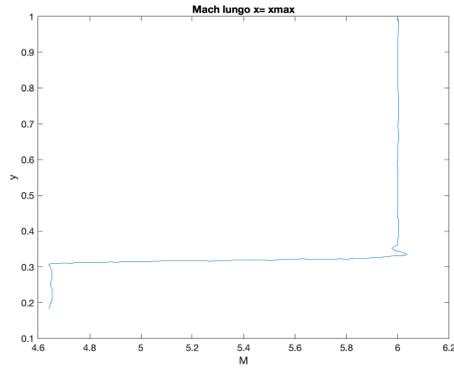
4.1.2 Mach = 6

La soluzione di riferimento è sempre ottenuta con SU2 con una risoluzione della mesh di $0.02 \times 0.02\text{mm}$ con 7500×5000 celle. L'onda d'urto si forma all'attacco della rampa alla posizione $x_s = 0.4879$ m. Come si può osservare nei grafico di figura 11b, il numero di Mach dopo l'urto arriva a $M = 5.7613$. L'onda avanza a $x_s = 0.549$ m per $y = 0.0405$ m e si estende fino a $y = 0.3097\text{m}$ (figure 11a). Si ottiene un coefficiente di resistenza pari a 0.01793 come da tabella 6.

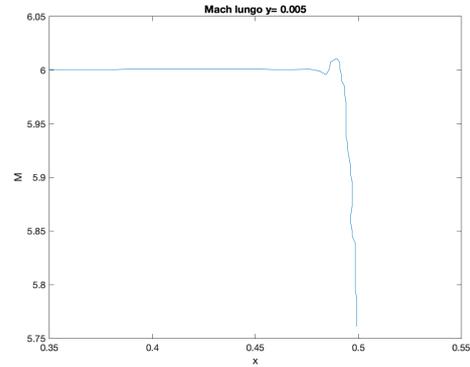
C_D		
SU2	Stoc.	$\Delta\%$
0.01793	0.01792	0.005

Tabella 6: Coefficiente di resistenza

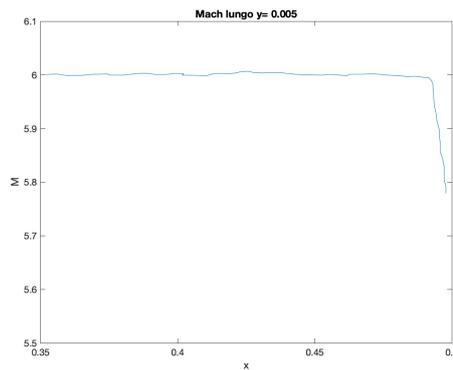
Il codice stocastico ottiene un coefficiente di resistenza con un errore del % rispetto alla soluzione di riferimento. Inoltre è interessante notare come l'oscillazione del numero di Mach nel campo di moto è molto più contenuta in direzione x a fronte di un infittimento del numero di celle che è stato aumentato del 12% rispetto al caso precedente della rampa con il flusso a Mach 2. La posizione dell'onda d'urto è ben catturata e si registra formarsi a $x_s = 0.4906$ m per la stazione $y = 0.005\text{m}$ (figura 11c).



(a) Rampa simulata con SU2. Spessore massimo dell'area a bassa velocità lungo $x_{max} = 1.5m$



(b) Rampa simulata con SU2. Numero di Mach prima e dopo l'urto lungo $y = 0.005m$



(c) Rampa simulata con il codice Stoc. Le oscillazioni del numero di Mach diminuiscono rispetto al caso $M=2$ a fronte di un aumento del numero di celle ($y = 0.005m$)

Figura 11: Rampa, risultati ottenuti con SU2 (11a e 11b) e con il codice Stoc. (11c)

4.2. Lastra

Si riporta per completezza il raggiungimento della convergenza di griglia per la soluzione di SU2 nel grafico in figura 12: si osserva che la differenza della soluzione riportata per la mesh $m3$ e $m4$ è inferiore allo 0.1% per quanto riguarda la posizione dell'onda d'urto. Considerando quindi l'indipendenza della soluzione dal numero di elementi di griglia sufficiente, si utilizza la griglia $m3$. Il massimo spessore della zona a basso numero di Mach dopo la lastra raggiunge un'altezza pari a $y_s = 0.305$ m in corrispondenza di $x = 1.5$ m (figura 13b). Il coefficiente di resistenza calcolato in questo caso è pari a $C_D = 0.197216$. Il codice stocastico in questo caso richiede un elevato numero di particelle per cella per evitare il rischio di avere celle pressoché vuote a causa dell'impatto con la lastra visto l'elevata velocità del campo di moto. Per questo motivo, se nei casi precedenti una media di 100 particelle per cella era sufficiente per avere una coerente valutazione della posizione dell'urto, qui non sono sufficienti. Dal momento che le celle situate sotto la lastra risultavano per lo più vuote dopo 100 step nonostante l'iniezione continua di particelle nel dominio, sono state inserite 4500 particelle per cella all'inizio di ogni time-step. Si osserva l'andamento del numero di Mach nel grafico di figura 13a confrontato con la soluzione di SU2. La soluzione risulta essere pienamente soddisfacente e in linea con la soluzione di SU2. Sebbene l'iniezione delle particelle anche dalla parete di destra aiuterebbe a compensare il problema delle celle meno popolate in seguito alla fase di avvezione, questa è stata disattivata per andare verso una diminuzione del tempo di calcolo. In questo caso dunque si attribuiscono alle particelle in uscita solo i valori di velocità peculiari più elevati secondo la distribuzione maxwelliana, eliminando del tutto i valori più bassi e meno probabili ma considerando l'alta velocità raggiunta a Mach $M = 6$, questo non risulta essere una forzatura in quanto la coda delle velocità inferiori tende rapidamente allo zero. Avendo deciso di accettare un tempo di calcolo inferiore alle 5 ore, sono state valutate diverse combinazioni tra numero di particelle per cella e numero di celle del dominio e per questo è proposta una risoluzione di 200×200 . Valutando la differenza di quantità di moto accumulata dalle particelle ad ogni passo temporale, con un appropriato raggruppamento si raggiunge un coefficiente di resistenza pari a $C_D = 0.196881$ a fronte di una differenza di quantità di moto complessiva lungo x pari a $0.9533 \frac{kgm}{s}$. Il risultato sul coefficiente rispetto al precedente calcolato con SU2 equivalente al 0.17%. L'errore inizia ad essere più significativo per la difficoltà nel mantenere un adeguato numero di particelle in ciascuna cella.

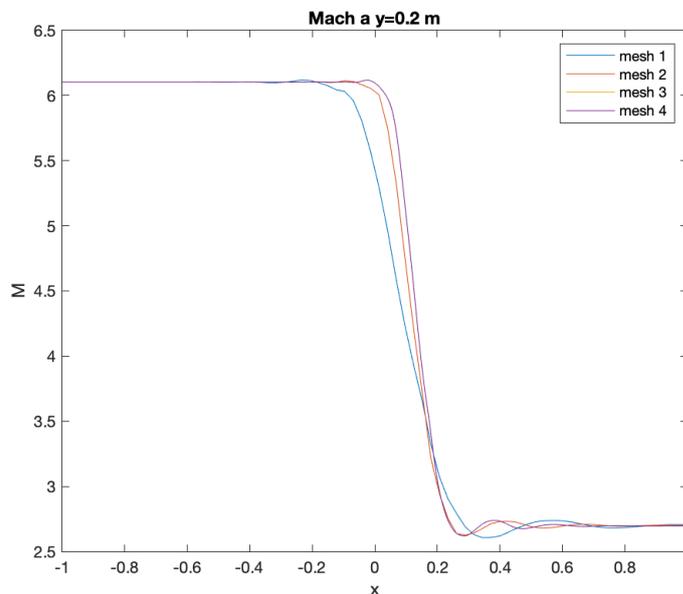
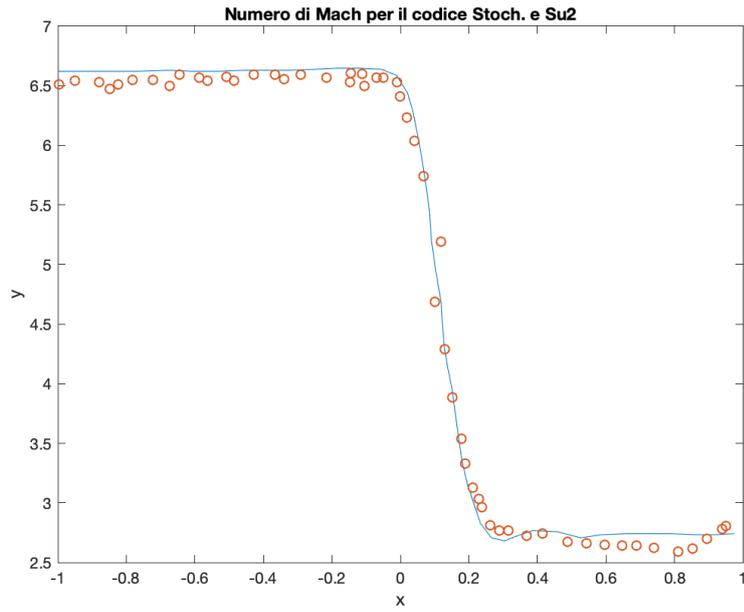
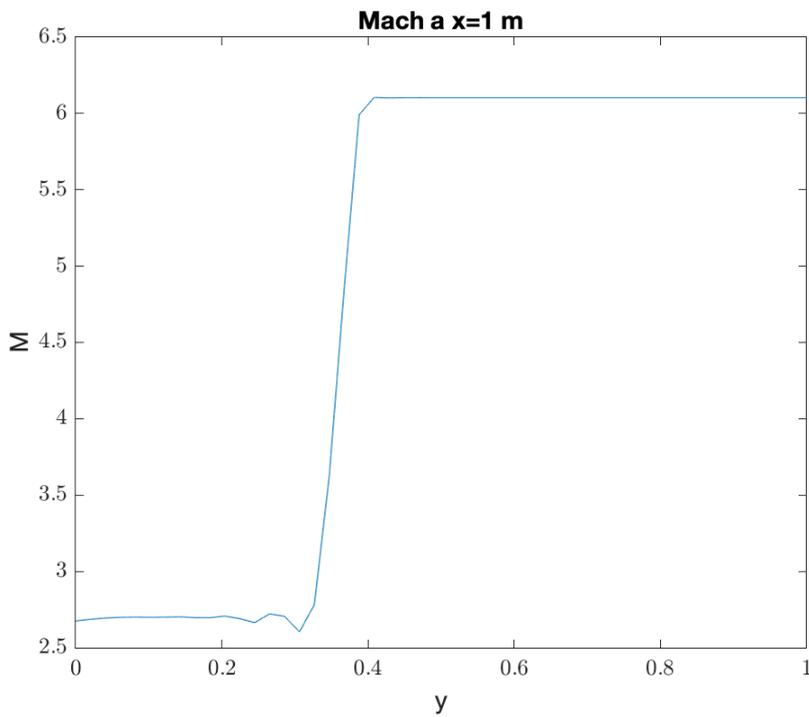


Figura 12: Lastra simulata con SU2. La convergenza di griglia è ritenuta raggiunta con la mesh $m3$.



(a) Lastra. I risultati in rosso del codice Stoc. catturano lo shock con un errore massimo inferiore allo 0.08% rispetto ai risultati ottenuti con SU2 (curva blu)



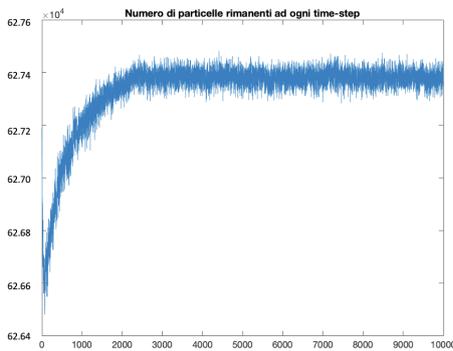
(b) Lastra simulata con SU2. Massimo spessore della zona a basso numero di Mach lungo $x_{max} = 1.5$ m

Figura 13: Lastra piana, risultati ottenuti con SU2 (13b) e con il codice Stoc 13a)

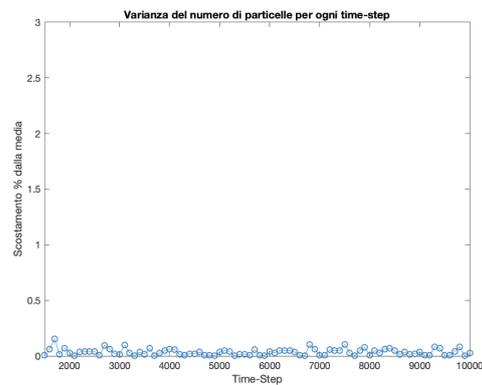
4.3. Flusso intorno ad uno ostacolo fisso

4.3.1 Flusso intorno ad uno ostacolo fisso $M=0$

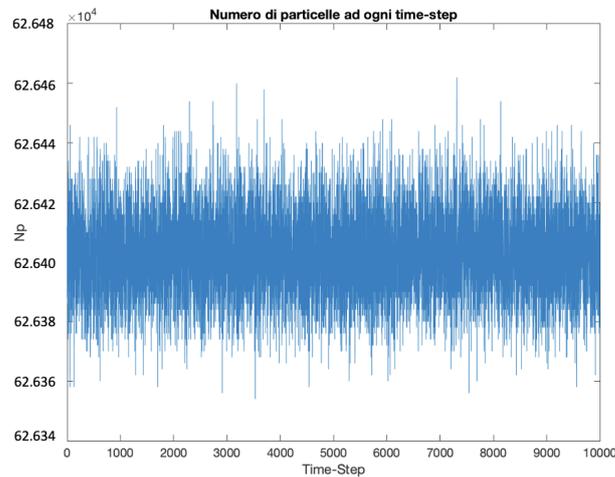
Questo problema viene simulato con il solo metodo Stoc. per soffermarsi sul raggiungimento della condizione di equilibrio. Il numero di particelle nel dominio raggiunge l'equilibrio dopo cinquemila time-steps. In particolare, a parità di numero di celle, un maggiore numero di particelle per cella garantisce l'equilibrio con un numero di steps inferiori. Si osserva infatti che la differenza di quantità di moto accumulata dalle particelle che impattano con il corpo rigido, dallo step di equilibrio $step - 0$ subisce una variazione che resta inferiore del 2% rispetto a quella dello step precedente. Le forze che il corpo esercita sul flusso nelle due direzioni si annullano a meno del rumore statistico e in particolare il valore della forza esercitata da ciascuna superficie è prossima alla pressione ambiente propria del flusso indisturbato. In figura 14c e 14b si osservano i due grafici che mostrano il numero di particelle nel dominio ad ogni time-step e la varianza in termini di scostamento percentuale dal numero medio di particelle presenti.



(a) Flusso $M=0$ intorno ad un ostacolo, $Np_C = 18$. Numero netto di particelle in ogni time-step raggiunge l'equilibrio dopo 3000 steps



(b) Flusso $M=0$ intorno ad un ostacolo, $Np_C = 18$. Scostamento dal valor medio di particelle nel dominio



(c) Flusso $M=0$ intorno ad un ostacolo, $Np_C = 50$. Numero di particelle all'inizio di ogni time-step

Figura 14: Simulazioni del flusso intorno ad un ostacolo con il codice Stoc all'equilibrio, $M=0$

Di seguito il grafico 15 mostra il numero di particelle che permangono nel dominio dopo la fase di avvezione in cui le particelle spostandosi possono trovarsi in cinque situazioni distinte:

- impattare con il corpo rigido e restare nel dominio
- impattare con il corpo rigido e raggiungere le superfici aperte di ingresso o uscita e quindi sono eliminate
- continuare il loro moto indisturbate

- impattare con la superficie superiore o inferiore e restare nel dominio
- impattare con la superficie superiore o inferiore e raggiungere le superfici di ingresso o uscita e vengono quindi eliminate

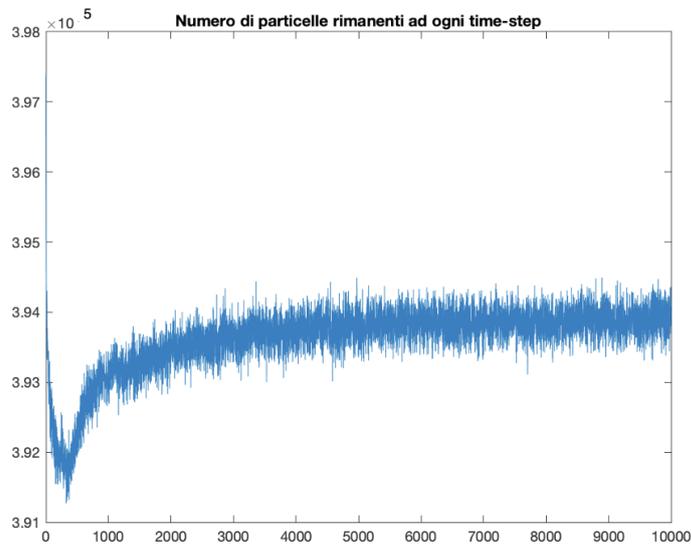


Figura 15: Flusso $M=0$ intorno ad un ostacolo, $Np_C = 50$. Il numero netto di particelle in ogni time-step raggiunge l'equilibrio dopo 1000 steps

I grafici riportano condizioni con un numero differente di particelle per cella. Il raggiungimento dell'equilibrio in termine di equilibrio delle forze sviluppate mostrano che il codice è stato implementato in maniera consistente con il principio di conservazione della quantità di moto. In questa simulazione il numero delle celle disponibili ad accogliere le particelle è pari a 34800 e la simulazione è durata in ore 1.38, 3.84, 7.68 per $Np_c = 18$, $Np_c = 50$, $Np_c = 100$ rispettivamente; il tempo di calcolo come atteso è linearmente proporzionale con il numero di particelle presenti nel dominio. Per avere una prima stima del tempo di calcolo, sono state precedentemente eseguite altre simulazione dello stesso problema ma con una griglia molto lasca per stimare l'effetto del numero di particelle sul tempo di esecuzione: l'andamento è mostrato nel grafico di figura16 ed è adimensionalizzato rispetto al numero di celle; si osserva che per numero di particelle elevate l'andamento risulta pressoché lineare. Sono state fatte prove per un basso numero di particelle per cella (da 18 a 40) e poi per un numero più elevato da 50 a 100.

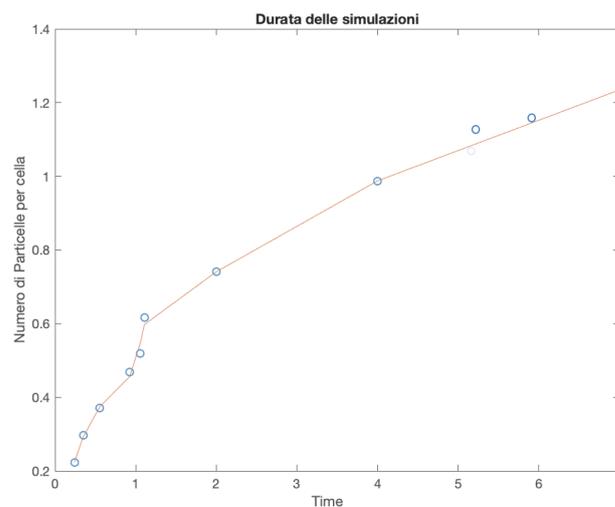


Figura 16: Durata delle simulazioni all'aumentare del numero di particelle per cella

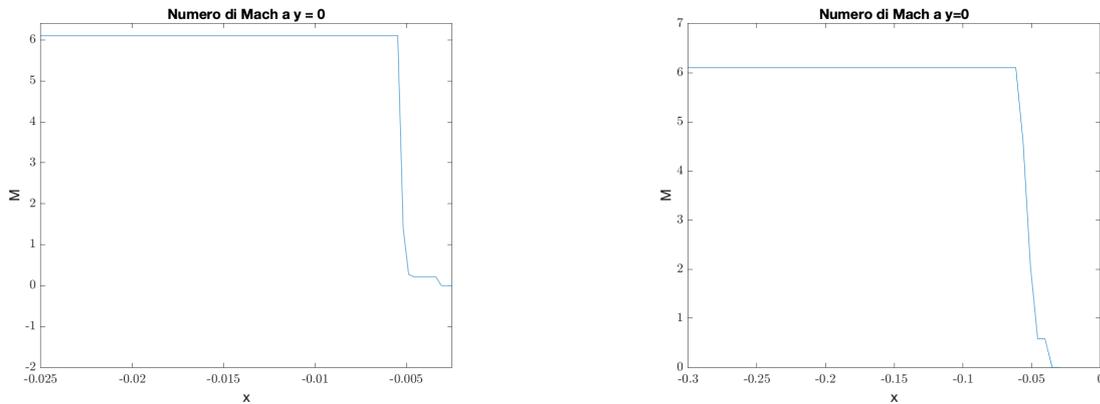
La variazione di quantità di moto accumulata dalle particelle che impattano con il corpo rigido inizia a stabilizzarsi rispetto allo step precedente intorno allo step identificato come step di equilibrio. La differenza di quantità di moto dovuta all'impatto da parte di ciascuna particella viene raccolta e accumulata per tutte le particelle all'interno del medesimo step: in questo modo è possibile ottenere la forza esercitata dalle quattro superfici di interesse aggiungendo una routine di raggruppamento dei dati su cui adoperare la media statistica. In particolare sono stati raccolti i valori ogni 100 time-step a partire dello step d'equilibrio e ciascuno di questi valori è a sua volta ottenuto mediandolo con sei cinque dati precedenti raccolti con frequenza pari a 2 step. Si osserva che la forza netta lungo le due direzioni $x - y$ non è perfettamente nulla ma presenta uno scostamento dallo zero pari allo 0.6% dovuto al rumore statistico. La tabella tab.7 riporta i valori della forza netta lungo x e lungo y per $Np_c = 100$.

$$\begin{vmatrix} F_{y1} & F_{x2} & F_{y3} & F_{x4} \\ -5399.1 \text{ N} & -5068.8 \text{ N} & 5420.9 \text{ N} & 5038.4 \text{ N} \end{vmatrix}$$

Tabella 7: Valori delle Forze per la simulazione M=0

4.3.2 Flusso a Mach 6 intorno ad un ostacolo fisso

I risultati prodotti da SU2 mostrano l'onda d'urto di prua che si genera in $\bar{x} = 0.0380$ nel caso di Sim. 1 dove il corpo 0.0025×0.0025 si trova a ridosso della superficie di uscita e a $\bar{x} = 0.06048\text{m}$ in Sim. 2 e Sim.3 (con una differenza dello 0.1% per la diversa discretizzazione spaziale del problema) dove il corpo si trova a distanza di 0.5 e 5m dalla superficie di uscita rispettivamente nei due casi. Si genera dunque una bolla subsonica a monte del corpo che nei tre casi risulta essere alla distanza $\Delta = 0.0355$ a meno di una tolleranza inferiore allo 0.1%; questa si può osservare nel grafico in figura 17a per quanto riguarda Sim. 1 e in figura 17b per Sim. 2 e Sim. 3.

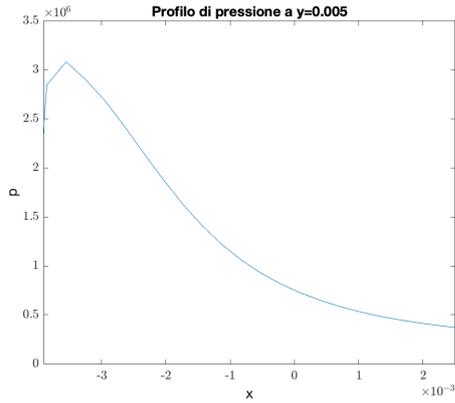


(a) SU2: Flusso intorno ad un ostacolo, M=6. Onda d'urto di prua lungo $y = 0$ per Sim.1

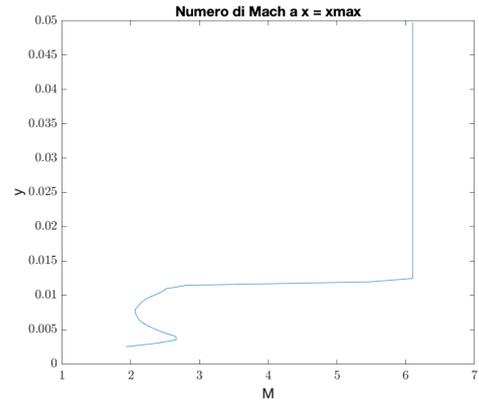
(b) SU2: Flusso intorno ad un ostacolo, M=6. Onda d'urto di prua lungo $y = 0$ per Sim.3

Figura 17: SU2: Onda d'urto di prua per Sim 1 e Sim 3

Per valori di y inferiori a $y = 0.005$ si osserva che si genera un ventaglio di espansione al vertice nord-ovest del corpo rigido come rappresentato dal grafico della pressione in figura 18a relativamente a Sim.1: la pressione, dopo l'aumento subito a causa dell'onda d'urto di prua, diminuisce con continuità e viste le dimensioni di Sim. 1, non raggiunge nuovamente le condizioni proprie del flusso indisturbato. Si riporta anche la variazione del numero di Mach lungo la superficie di uscita in figura 18b relativa a Sim.1 e si può osservare coerentemente con quanto detto sopra la presenza della zona di espansione nella regione immediatamente superiore al corpo. Per $y > 0.01\text{m}$ il flusso si trova esterno all'onda d'urto e il campo di moto è quello del flusso indisturbato.



(a) SU2: Flusso intorno ad un ostacolo, $M=6$. Andamento della pressione nel ventaglio d'espansione a $y=0.005$ m, Sim 1



(b) SU2: Flusso intorno ad un ostacolo, $M=6$. Spessore della zona a bassa velocità lungo $x = x_{max}$, Sim 1

Figura 18: SU2: Andamento della pressione del numero di Mach per sim 1

Il grafico di figura 19 mostra che la scia a valle del corpo non è ben catturata con la superficie in uscita a $x_{max} = 0.3$ infatti il numero di Mach lungo la coordinata y ha una crescita molto lenta per y compresa tra 0 e 0.25 prima di uscire dalla regione perturbata e tornare a Mach $M = 6$ nel campo di moto indisturbato con una discontinuità repentina tra $y = 0.24$ e $y = 0.28$ m.

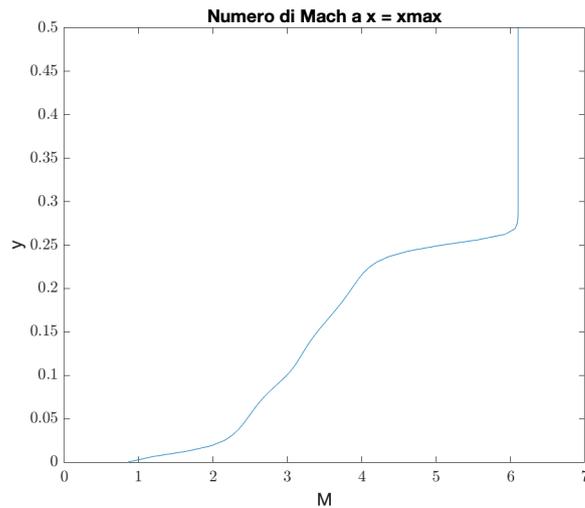
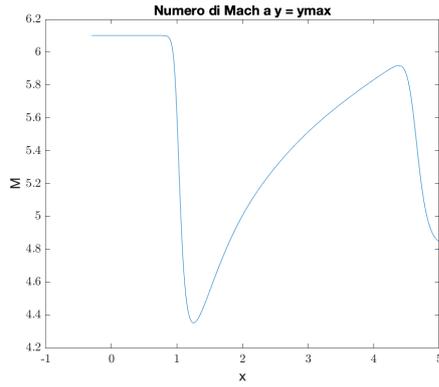
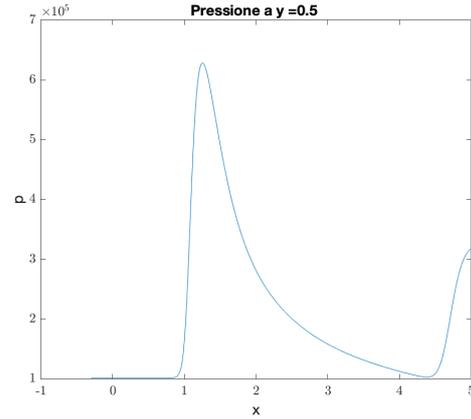


Figura 19: SU2: Flusso intorno ad un ostacolo, $M=6$. Andamento del numero di Mach lungo $x = x_{max}$ considerando anche la scia - Sim.2

Le informazioni aggiuntive ricavate da Sim.3 sono l'andamento del campo di moto a valle del corpo rigido mentre l'onda d'urto si riflette sul contorno superiore del dominio e si alternano zone ad alta e bassa velocità come si può osservare dall'andamento di numero di Mach e pressione in figura 20a e 20b.



(a) SU2: Flusso intorno ad un ostacolo, $M=6$. Numero di Mach a $y = 0.5$ - Sim.3 riflessione a valle del corpo rigido



(b) SU2: Flusso intorno ad un ostacolo, $M=6$. Pressione lungo $y = 0.5$ - Sim.3 riflessione a valle del corpo rigido

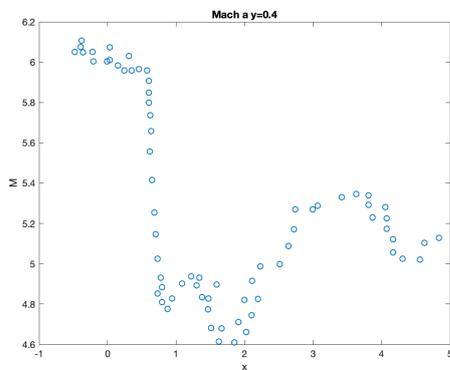
Figura 20: SU2: Flusso intorno ad un ostacolo, $M=6$. Andamento della pressione del numero di Mach per sim 3

Per concludere si riportano i coefficienti di resistenza ottenuti per confrontarli con quelli del calcolo stocastico in tabella 8. La simulazione arriva all'equilibrio circa il numero di particelle dopo 10000step in tutte e tre le configurazioni e viene fatta durare per 100000 intervalli in totale.

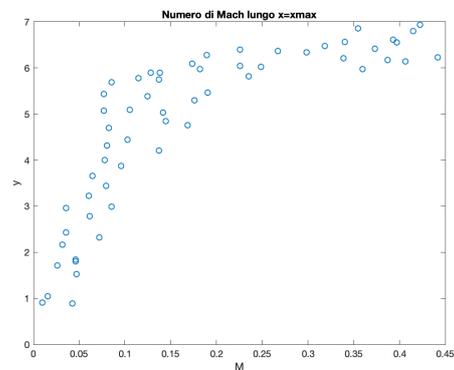
	SU2	Codice Stoc.	$\Delta\%$
Sim.1	0.004377	0.00436	0.3
Sim.2	0.043167	0.04297	0.45
Sim.3	0.037237	0.03721	0.07

Tabella 8: Coefficienti di resistenza per Sim.1, Sim.2, Sim.3

Si osserva dal grafico nell'immagine 22a, 22b, 22c che lo shock prima del corpo è ben catturato in tutte e tre le simulazioni mentre allontanandosi da esso i risultati sono affetti da diffusione numerica soprattutto nel caso di Sim.3 per y maggiori di $y = 0.09$ (figura 21a). Nel caso Sim. 2, il numero di Mach lungo la sezione di uscita torna pari a $M = 6$ come nelle condizioni di flusso indisturbato ma l'andamento non segue il profilo precedentemente mostrato nella soluzione di riferimento (21b). Si osserva dunque un valore di coefficiente di resistenza inferiore rispetto al caso di SU2 con un errore significativo pari allo 0.45%.

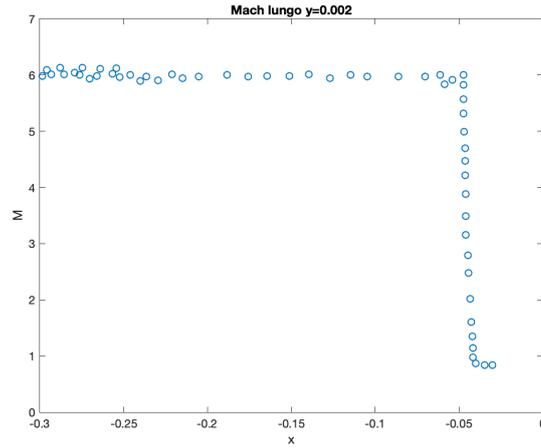


(a) Numero di Mach lungo $y = 0.4$ - Sim.3

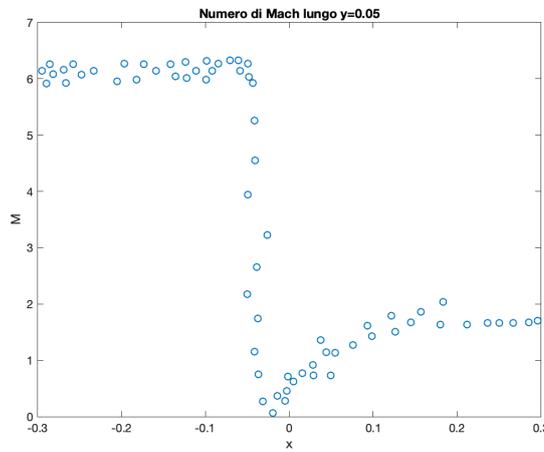


(b) Numero di Mach per $x=x_{max}$, Sim.2

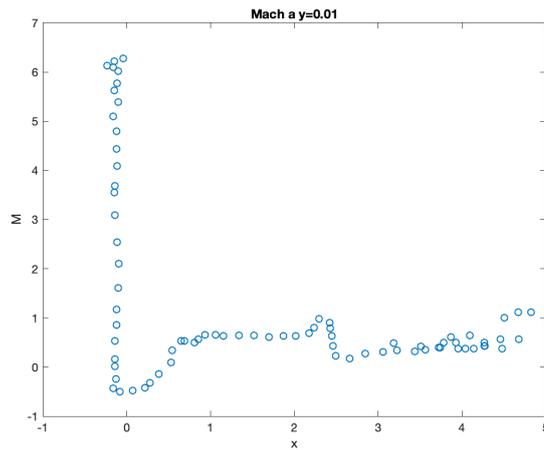
Figura 21: Stoc: Numero di Mach



(a) Codice Stoc: Flusso intorno ad un ostacolo, $M=6$.
 Numero di Mach lungo $y = 0.002$ - Sim.1



(b) Codice Stoc: Flusso intorno ad un ostacolo, $M=6$.
 Numero di Mach lungo $y = 0.05$ - Sim.2



(c) Codice Stoc: Flusso intorno ad un ostacolo, $M=6$.
 Numero di Mach lungo $y = 0.01$ - Sim.3

Figura 22: Codice Stoc: Flusso intorno ad un ostacolo, $M=6$. Shock capturing

5. Conclusioni e sviluppi

Dopo l'analisi delle tre principali simulazioni riportata, si può stabilire che il metodo proposto riesce a soddisfare l'esigenza di valutare il coefficiente di resistenza nei problemi descrivibili dalle equazioni di Eulero. La simulazione più difficoltosa da gestire, è stata la simulazione della lastra piana a $M = 6$ questo perché l'efficienza del codice stocastico implementato, è molto sensibile alle dimensioni del problema e alle scale di grandezza che lo caratterizzano. Nel caso della lastra infatti, il primo test portava ad ottenere dei valori di velocità delle celle NaN : le grandezze macroscopiche sono tutte ottenute come rapporto con il numero di densità proprio della cella. Dopo un diversi tentativi si è osservato che mediamente allo step 200, in seguito alla fase di avvezione, molte celle restavano vuote vista l'elevata velocità del flusso medio. Questa situazione rischia di verificarsi frequentemente se il design dell'esperimento non è ben ragionato. Il problema sta nel fatto che una prima strategia è diminuire l'ampiezza dell'intervallo temporale: sotto un certo livello in poi però, questa tecnica rende completamente inutile la presenza della routine di iniezione delle particelle nel dominio. Le celle che soffrono per prime di questa problematica sono quelle in prossimità delle pareti: da un lato ingrandire le dimensioni delle celle diminuirebbe la probabilità di celle vuote, ma dall'altro le grandezze macroscopiche non sarebbero più rappresentative. Bisogna anche considerare che se è presente un ostacolo nel dominio, aumentare le dimensioni delle celle deve garantire comunque la possibilità di avere una buona risoluzione anche nella discretizzazione dell'ostacolo. Ed è questo che ha reso inutilizzabile il primo test fatto sulla lastra piana. Aumentare le dimensioni dell'ostacolo a parità di griglia permette una migliore risoluzione ma aumenta anche il rumore statistico. In aggiunta, è quasi sempre necessario anche aumentare le dimensioni del dominio per ottenere delle soluzioni fisicamente accettabili e questo, anche a parità di dimensione della cella, implica comunque un aumento del numero di celle. Dal momento che il numero di particelle medio è stabilito a partire dal parametro Np_c , l'aumento del numero di celle -anche minimo- comporta ad un aumento dei tempi di calcolo che penalizza la potenzialità del codice. D'altra parte utilizzare meno di dieci particelle per cella farebbe perdere di validità fisica alla soluzione ottenuta. Risulta evidente la necessità di ottimizzare il codice con delle griglie di dimensione variabile lungo le direzioni di interesse e anche attribuire il parametri Np_c alla specifica cella j_c e non mantenerlo omogeneo in tutto il dominio. Questa nuova via andrebbe implementata costruendo una variabile tipo 'cella' caratterizzata dalle coordinate del suo centro x_c, y_c , dal suo indice j , dal suo dx, dy e Np_c . In questo modo la routine che si occupa della collisione non ha bisogno di particolari modifiche ma può continuare a definire le collisioni tra particelle nello spazio delle velocità a partire dall'indice della cella j con un semplice ciclo 'DO' accedendo all'informazione di Np_c per quella cella j . Una modifica di questo tipo per esempio, permetterebbe di simulare il caso del corpo rigido con un numero di particelle maggiore nelle celle in prossimità del corpo stesso. La simulazione fatta con il flusso intorno al corpo rigido mostra che raggiunge l'equilibrio delle forze per $M = 0$ e questo è un punto di partenza fondamentale per attribuire al corpo almeno il grado di libertà della traslazione. In questo caso la velocità del corpo rigido può essere ottenuta a partire dalla conservazione della quantità di moto all'interno del dominio e quindi la differenza di quantità di moto subita dalle particelle a causa della presenza del corpo, diventa facilmente trasformabile nel decremento di velocità del corpo stesso. Anche questo caso si prospetta facilmente inseribile nel codice proposto: una volta aggiornata la nuova velocità del corpo rigido alla fine della fase di avvezione e compiuta la fase di collisione tra le particelle, il corpo può essere spostato. Diventa certamente fondamentale stabilire nuovamente a quale gruppo appartiene ciascuna cella all'inizio del time-step successivo: se è mista, interna o esterna; la condizione di riflessione speculare con le pareti del corpo dovrà tenere in considerazione il moto relativo corpo-particella. Certamente in questo caso il tempo di calcolo aumenta per la necessità di intensificare la risoluzione e di ingrandire il dominio per meglio rappresentare le condizioni delle onde d'urto. Un intervento radicale e una riscrittura del codice in parallelo potrebbe essere necessaria. La grande potenzialità di questo metodo con tutte le odierne limitazioni, sta nella certezza che il calcolo arriverà ad un risultato e se la implementazione dei fenomeni fisici di interesse sono ben descritti, non c'è bisogno di forzare nessuna condizione per la tipologia dei problemi analizzati in questo lavoro senza nemmeno la necessità di dover seguire le traiettorie di ciascuna particella. Si è visto che globalmente, l'incipienza dell'onda d'urto è stata sempre visualizzata con un errore non superiore all'1% così come per il coefficiente di resistenza, questo si considera un ottimo risultato.

A. Appendice: Mappe 2D

Per completezza si riportano le mappe 2D per i casi più significativi che sono stati simulati dal codice Stoc implementato. In particolare si trova la mappa relativa alla rampa simulata con flusso $M = 2$ (Sezione 4.1) che il codice è ben in grado di rappresentare in termini di distribuzione del numeri di Mach (figura 23).

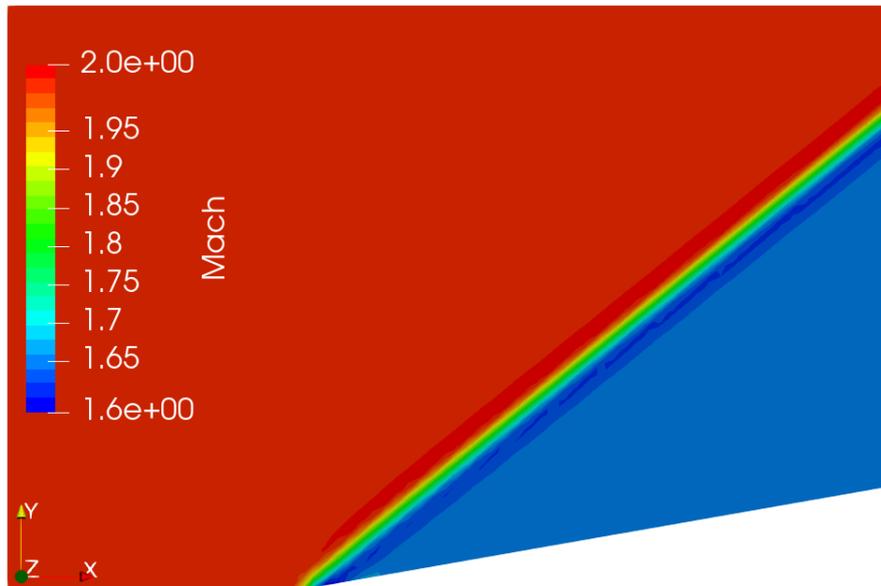
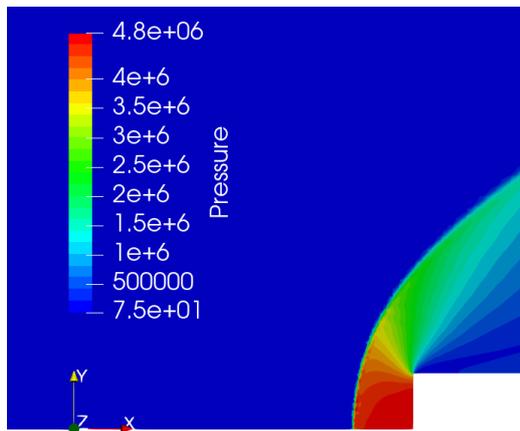
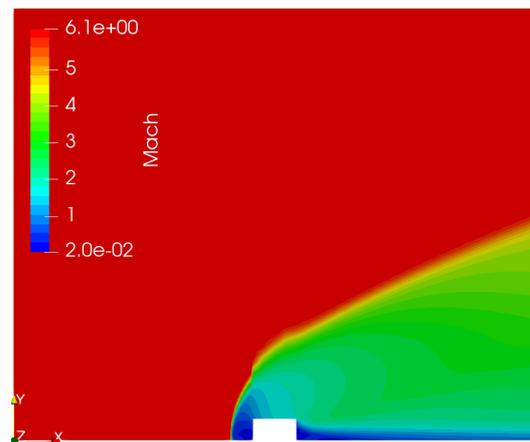


Figura 23: Metodo stocastico. Rampa $M = 2$: Mappa 2d, numero di Mach

Si mostrano inoltre le mappe della pressione per quanto riguarda il flusso intorno all'ostacolo a $M = 6$ (Sezione 4.3) di Sim 1 in figura 24a; in figura 24b si osserva la distribuzione del numero di Mach e il flusso anche a valle dell'ostacolo (sim 2).



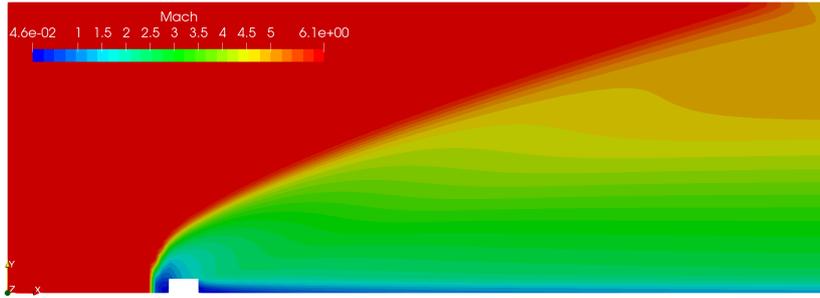
(a) Flusso intorno ad uno ostacolo: Mappa 2D della pressione a monte dell'ostacolo, Sim.1



(b) Flusso intorno ad uno ostacolo: Mappa 2D del numero di Mach, Sim.2

Figura 24: Metodo stocastico. Mappa 2D della pressione in Sim.1 (24a) e del numero di Mach per Sim.2 (24b)

La figura 25 riporta la scia e la riflessione dell'onda d'urto di prua sulla parete superiore del dominio. Il campo di moto non riesce ad essere ben descritto dal codice stocastico. Per questo ultimo set di simulazioni, le mappe sono prodotte solo per la parte superiore del dominio sfruttando per semplicità la simmetria del problema.



(a) Flusso intorno ad uno ostacolo: Mappa 2D del numero di Mach, dettaglio Sim.3



(b) Flusso intorno ad uno ostacolo: Mappa 2D del numero di Mach, riflessione dell'urto di prua Sim.3

Figura 25: Metodo stocastico. Sim 3: zoom sul corpo rigido con l'onda d'urto di prua (25a), riflessione dell'onda d'urto a valle del corpo rigido

Abstract in lingua italiana

Partendo dall'ampio impiego del metodo DSMC nell'ambito dei gas rarefatti, l'interesse del lavoro è quello di proporre una rielaborazione di tale metodo di simulazione per la risoluzione del problema di Eulero nel caso di un flusso a temperatura e pressione ambiente quando in esso si genera un'onda d'urto a causa dell'interazione del campo di moto con un ostacolo. L'obiettivo è proporre un metodo che richieda poco sforzo nei casi in cui si vuole essere in grado di stimare il coefficiente di resistenza globale che si genera su un ostacolo rappresentabile come un corpo rigido o su una parete solida all'interno del dominio di calcolo. Il metodo implementato è scritto in linguaggio Fortran e si basa su due parametri di input fondamentali: il numero e la dimensione delle celle che costituiscono la griglia e il numero di particelle in ciascuna cella. Il metodo non segue la traiettoria delle singole particelle dal momento che i casi simulati presentano un alto numero di densità, ma introduce il parametro di peso che si attribuisce alle particelle in base alla cella cui appartengono e il peso è l'unico parametro utile alla trasformazione delle variabili numeriche in variabili fisiche. Di fatto il lavoro è incentrato sulla programmazione del corpo principale del metodo di calcolo ossia sull'inizializzazione delle particelle e la generazione delle velocità peculiare secondo la funzione di distribuzione di probabilità di Maxwell, l'avvezione e l'interazione delle particelle con le superfici solide presenti nel dominio e la collisione tra le particelle. Questo step del lavoro è stato affiancato dal continuo controllo dei test fatti tramite le soluzioni fornite da un software di riferimento identificato in SU2. Tutto questo si colloca nella volontà di fornire un codice che possa essere ulteriormente sviluppato in seguito per simulare un flusso intorno ad un corpo rigido in movimento il cui moto è descritto dalle equazioni di Newton e Eulero come risultato del trasferimento di quantità di moto da parte delle particelle di flusso che urtano il corpo.

Parole chiave: DSMC, pdf , collisione, onda d'urto, SU2

Riferimenti bibliografici

- [1] Federico Bariselli, Aldo Frezzotti, Annick Hubin, and Thierry Magin. Aerothermodynamic modelling of meteor entry flows. In *Monthly Notices of the Royal Astronomical Society*, volume 492, pages 2308–2325, 2019.
- [2] Graeme A Bird. Molecular gas dynamics and the direct simulation of gas flows. *Molecular gas dynamics and the direct simulation of gas flows*, 1994.
- [3] Thomas D Economou, Francisco Palacios, Sean R Copeland, Trent W Lukaczyk, and Juan J Alonso. Su2: An open-source suite for multiphysics simulation and design. *Aiaa Journal*, 54(3):828–846, 2016.
- [4] Christophe Geuzaine and Jean-François Remacle. Gmsh: A 3-d finite element mesh generator with built-in pre-and post-processing facilities. *International journal for numerical methods in engineering*, 79(11):1309–1331, 2009.
- [5] Randall J LeVeque and Randall J LeVeque. *Numerical methods for conservation laws*, volume 214. Springer, 1992.
- [6] Charles R Lilley and Michael N Macrossan. Methods for implementing the stream boundary condition in dsmc computations. *International journal for numerical methods in fluids*, 42(12):1363–1371, 2003.
- [7] M.N Macrossan. The equilibrium flux method for the calculation of flows with non-equilibrium chemical reactions. *Journal of Computational Physics*, 80:204–231, 1989.
- [8] Dieter Mewes and Franz Mayinger, editors. *Direct Simulation Monte-Carlo (DSMC) Method*, pages 275–315. Springer Berlin Heidelberg, Berlin, Heidelberg, 2005.
- [9] D.I Pullin. Direct simulation methods for compressible inviscid ideal-gas flow. *Journal of Computational Physics*, 34:231–244, 1980.

[10] P.A. Thompson. *Compressible-fluid Dynamics*. Advanced engineering series. 1988.

Acknowledgements

Si ringrazia primo tra tutti il Professor A. Frezzotti per la sua paziente disponibilità che non è venuta mai meno, in tutte le fasi di questo lavoro.

Si ringraziano in particolare alcune persone che sono state davvero fondamentali per arrivare a scrivere l'ultima riga di oggi: Sam e Chicca che sanno aspettarmi quando ho bisogno di prendere fiato rallentando il loro passo, ma anche traghettarmi nelle burrasche vigili e realisti (almeno Chicca..). Ringrazio Teo, ultimo ritrovato ma prezioso come pochi e fondamentale in queste ultime settimane di lavoro. Non posso fare a meno di ringraziare Cesare che non viene da questo contesto universitario ma è riuscito a diventare colonna portante anche qui, anche a 500 km di distanza. Per tutto questo grazie. C'è un riconoscente grazie anche per gli amici della Cascina, con la loro compagnia straordinaria. L'ultimo grazie immenso è per Monica e Filippo, un bel po' di tutto questo è per voi.