



Gasdinamica dei processi industriali

Corso di Laurea Magistrale in Ingegneria Energetica
Alunno: Valerio Pagliarella
Matricola - 0287800

*Analisi numeriche 2D per la soluzione di un flusso laminare
e incomprimibile mediante il metodo dei volumi finiti:
confronto tra algoritmi SIMPLE e SIMPLER*

Docenti:

Prof. Giacomo Falcucci

Dott. Vesselin Krassimirov Krastev

Indice

1. Introduzione
2. Algoritmi SIMPLE e SIMPLER
3. Risultati caso 1
4. Risultati caso 2
5. Caso non stazionario

1. Introduzione

Con questo report si intende studiare mediante il metodo dei volumi finiti (FVM) il moto di un fluido all'interno di due differenti configurazioni geometriche. L'analisi numerica è stata condotta in Matlab e i risultati ottenuti verranno confrontati con quelli ricavati dal software commerciale **Ansys Fluent**. Il fluido utilizzato per le simulazioni è l'**idrogeno atomico H**, le cui principali proprietà fisiche, selezionate dal database di Fluent, sono:

$$\rho = 0.040948 \frac{\text{kg}}{\text{m}^3} \quad (1.1)$$

$$\mu = 7.49 \cdot 10^{-6} \frac{\text{kg}}{\text{m} \cdot \text{s}} \quad (1.2)$$

- CASO 1: canale 2D

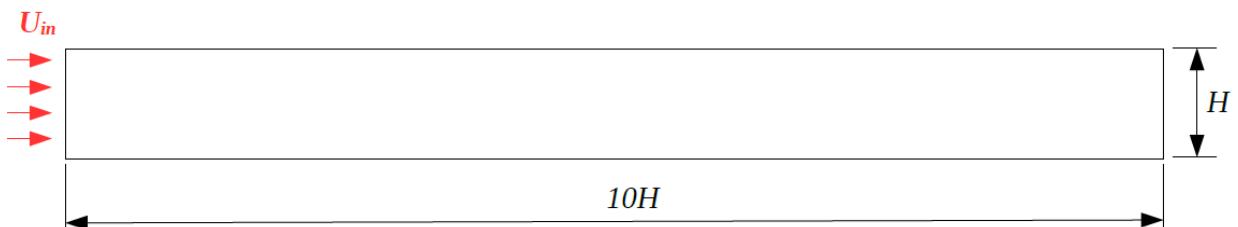


Figura 1: canale dritto

Il primo caso studia il flusso laminare di un fluido tra lastre piane parallele, con geometria del problema assegnata e nota la velocità del fluido in ingresso u_{inlet} attraverso il Re. Si richiede di confrontare la soluzione numerica stazionaria ottenuta in Matlab con il profilo teorico di Poiseuille. In particolare:

- a. Sono stati studiati due regimi di moto laminare, rispettivamente:

Re	50
Re	100

Tabella 1: valori Reynolds caso canale dritto

dove:

$$Re = \frac{U_{in} H}{v} \quad (1.3)$$

- b. Sono state utilizzate due griglie di calcolo differenti:

	n	m
Mesh 1	60	20
Mesh 2	120	40

Tabella 2: mesh canale dritto

Dove **n** ed **m** sono gli intervalli spaziali tra i nodi di pressione lungo la direzione x e lungo la direzione y rispettivamente.

- c. Vengono utilizzati due schemi di interpolazione per i termini convettivi: **upwind e differenze centrate**;
- d. Vengono confrontate le prestazioni del metodo **SIMPLE** con quelle del metodo **SIMPLER**.

- CASO 2: canale 2D con gradino

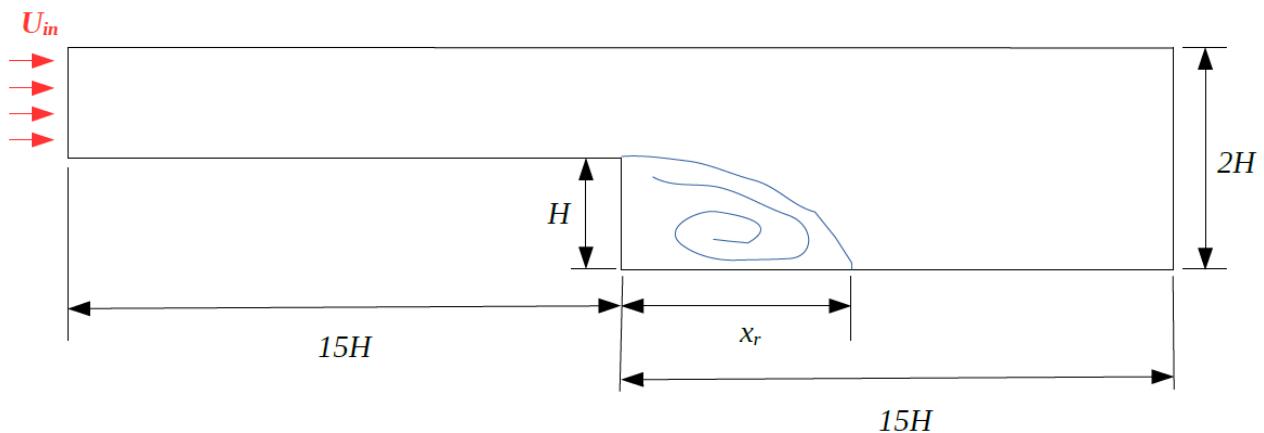


Figura 2: canale con gradino

In questa configurazione è presente un gradino a metà canale.

Si vogliono confrontare i valori della **lunghezza di riattacco** della vena fluida ottenuti in Matlab e in Fluent con i seguenti valori sperimentali:

Re	x_r/H
100	2.73
150	3.73
250	6.21

Tabella 3: valori sperimentali delle lunghezze di riattacco della vena fluida

Dove:

$$Re = \frac{U_{in} \cdot (2H)}{\nu} \quad (1.4)$$

Si vuole studiare inoltre il **problema non stazionario** modificando opportunatamente gli algoritmi SIMPLE e SIMPLER. Anche per questo caso:

- a. Sono state confrontate due griglie di calcolo:

	n	m
Mesh 1	60	20
Mesh 2	120	40

Tabella 4: mesh canale con gradino

- b. Sono stati utilizzati due schemi di interpolazione, **upwind** e **differenze centrate**;
- c. Viene effettuato il confronto tra algoritmo **SIMPLE** e **SIMPLER**.

2. Algoritmi SIMPLE e SIMPLER

Prima di mostrare le modalità di ottenimento dei risultati, si procede a riassumere le metodologie iterative utilizzate nei solutori (Matlab e Fluent) ovvero gli algoritmi SIMPLE e SIMPLER. Brevemente si richiamano le equazioni che permettono di risolvere la dinamica tridimensionale (in questo caso bidimensionale) di un flusso incomprimibile, ovvero la **relazione di continuità** e le **equazioni di Navier-Stokes**:

$$\operatorname{div} \mathbf{u} = 0 \quad (2.1)$$

$$\frac{\partial \mathbf{u}}{\partial t} + \operatorname{div}(\mathbf{u}\mathbf{u}) = -\frac{1}{\rho} \frac{\partial p}{\partial x} + \operatorname{div}(v \operatorname{grad} u) \quad (2.2)$$

$$\frac{\partial v}{\partial t} + \operatorname{div}(v\mathbf{u}) = -\frac{1}{\rho} \frac{\partial p}{\partial y} + \operatorname{div}(v \operatorname{grad} v) \quad (2.3)$$

Come noto, la soluzione di queste equazioni non è banale in quanto bisogna affrontare due problematiche in particolare:

1. la **non linearità dei termini convettivi**, che può essere risolta attraverso una forma linearizzata delle equazioni di partenza con cui approcciare diversi metodi iterativi di soluzione;
2. la presenza del gradiente di pressione comporta **l'accoppiamento** tra la pressione stessa e le componenti di velocità, complicando la discretizzazione del termine $\operatorname{grad}(p)$ e implicando un problema di inconsistenza della fisica descritta dalle equazioni discretizzate.

Per la discretizzazione delle equazioni di bilancio della quantità di moto, si possono allora sfruttare delle **griglie staggered**, spostando la valutazione delle componenti di velocità u e v , dal centro cella alle interfacce della stessa. In questo modo si aggira la questione legata all'accoppiamento pressione-velocità, discretizzando il termine $\operatorname{grad}(p)$ presente nelle equazioni di bilancio della quantità di moto sfruttando differenze di pressione tra nodi contigui e non alterni. Nel caso specifico, la componente di velocità u associata al generico nodo scalare viene allocata all'interfaccia del volume di controllo iniziale spostandola verso sinistra; analogamente per la componente di velocità v che viene allocata all'interfaccia del volume di controllo attraverso uno spostamento verso il basso. Di seguito è riportata una rappresentazione di una porzione generica della griglia di calcolo:

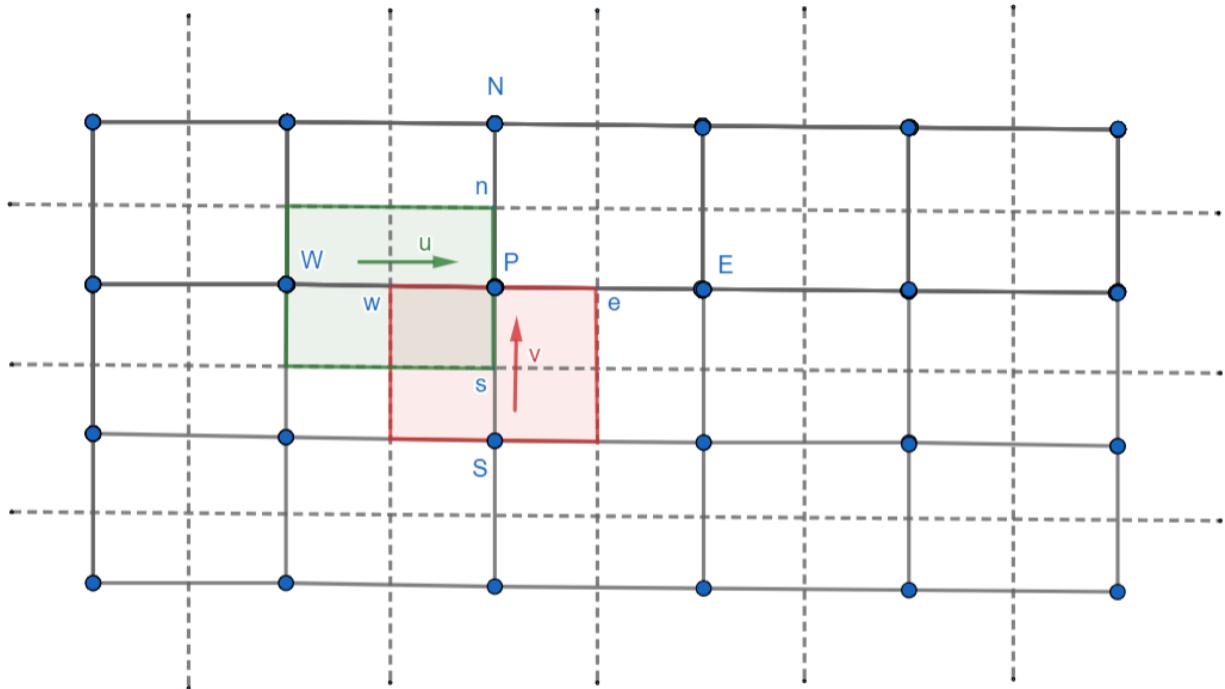


Figura 3: griglia staggered

Concentrandoci sulla griglia di calcolo per il campo di velocità u :

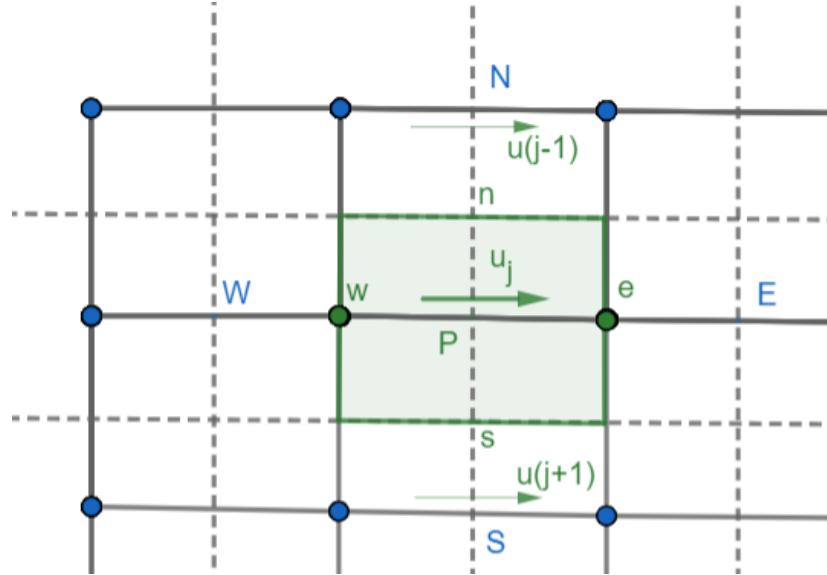


Figura 4: generica cella campo di velocità u

In questo modo la forma discretizzata dell'equazione di bilancio della quantità di moto lungo x si presenta come:

$$a_e = \sum_{nb} a_{nb} + \Delta F_{ew} + \Delta F_{ns} \quad (2.4)$$

I coefficienti a_e e a_{nb} includono i contributi di convezione e diffusione distinti in base allo schema di interpolazione utilizzato. Come accennato nel *paragrafo 2* sono stati utilizzati lo schema **upwind** e lo schema **differenze centrate**. L'idea alla base del secondo schema è quella di sostituire, ad ogni derivata che compare nell'equazione da approssimare, un rapporto incrementale finito. La principale mancanza dello schema differenze centrate è la sua incapacità di identificare la direzione del flusso. In presenza di un forte flusso convettivo, tale schema risulta impreciso. Lo schema upwind migliora questo aspetto essendo uno schema direzionale, tiene conto infatti della direzione del flusso nella determinazione del termine convettivo in corrispondenza dell'interfaccia della cella, a cui verrà assegnato un valore pari a quello del nodo immediatamente a monte rispetto al verso del flusso convettivo. A titolo di esempio, vengono riportati di seguito le forme dei coefficienti associati ai nodi di bulk del dominio per entrambi gli schemi:

a. Upwind QDM-x :

$$\begin{aligned} a_w &= [\max(F_{uw}, 0) + D_u] \cdot \Delta y \\ a_E &= [\max(0, -F_{ue}) + D_u] \cdot \Delta y \\ a_N &= [\max(0, -F_{vn}) + D_v] \cdot \Delta x \\ a_S &= [\max(F_{vs}, 0) + D_v] \cdot \Delta x \\ S_p &= - [(F_{ue} - F_{uw}) \cdot \Delta y + (F_{vn} - F_{vs}) \cdot \Delta x] \\ a_p &= a_E + a_w + a_N + a_S - S_p \end{aligned} \quad (2.5)$$

b. Differenze centrate QDM-x :

$$\begin{aligned}
 a_w &= (D_u + F_{uw}/2) \cdot \Delta y \\
 a_E &= (D_u - F_{ue}/2) \cdot \Delta y \\
 a_N &= (D_v - F_{vn}/2) \cdot \Delta x \quad (2.6) \\
 a_S &= (D_v + F_{vs}/2) \cdot \Delta x \\
 S_p &= - [(F_{ue} - F_{uw}) \cdot \Delta y + (F_{vn} - F_{vs}) \cdot \Delta x] \\
 a_p &= a_E + a_w + a_N + a_S - S_p
 \end{aligned}$$

Dove:

$$D_u = \frac{\mu}{\Delta x} \quad (2.7)$$

$$D_v = \frac{\mu}{\Delta y} \quad (2.8)$$

Una volta discretizzate le equazioni si procede con l'implementazione di un algoritmo iterativo di soluzione. Con l'algoritmo **SIMPLE** (*Semi-Implicit Method for Pressure-Linked Equations*), come prima iterazione, per risolvere l'equazione 2.4 si ipotizza una distribuzione di pressione iniziale p^* : si ricava tuttavia un campo di velocità imperfetto in quanto non soddisfa la continuità. Per ottenere i campi corretti bisogna tener conto di alcuni termini correttivi:

$$p = p^* + p' \quad (2.9)$$

$$u = u^* + u' \quad (2.10)$$

$$v = v^* + v' \quad (2.11)$$

Il legame tra le correzioni di velocità e pressione è dato dall'equazione:

$$a_e u_e' = \sum_{nb} a_{nb} u_{nb}' + \frac{(p_p' - p_e')}{\rho} A_e \quad (2.12)$$

Che può essere riscritta in forma più compatta trascurando il primo termine a secondo membro:

$$u_e' = d_e + (p_p' - p_e') \quad (2.13)$$

e analogamente:

$$v_n' = d_n + (p_p' - p_n') \quad (2.14)$$

con:

$$d_e = \frac{A_e}{\rho a_e} \quad (2.15)$$

$$d_n = \frac{A_n}{\rho a_n} \quad (2.16)$$

Con queste equazioni è possibile allora calcolare i campi corretti di velocità e pressione:

$$u = u^* + d_e (p_p' - p_e') \quad (2.17)$$

$$v = v^* + d_n (p_p' - p_n') \quad (2.18)$$

Per ricavare un'equazione discretizzata per il campo p' si può sfruttare l'equazione di conservazione della massa nel caso bidimensionale:

$$\frac{\partial \rho}{\partial t} + \frac{\partial(\rho u)}{\partial x} + \frac{\partial(\rho v)}{\partial y} = 0 \quad (2.19)$$

Dall' integrazione dell'equazione 2.19 rispetto il volume di controllo attorno il generico nodo di pressione P si ottiene la corrispettiva forma discretizzata:

$$\frac{(\rho_p - \rho_p^0) \Delta x \Delta y}{\Delta t} + [(\rho u)_e - (\rho u)_w] \Delta y + [(\rho u)_n - (\rho u)_s] \Delta x = 0 \quad (2.20)$$

Per il caso stazionario e incomprimibile, per sostituzione delle equazioni 2.17 e 2.18 nella 2.20 si ottiene:

$$a_p p'_p = a_E p'_E + a_w p'_w + a_n p'_n + a_s p'_s - b^* \quad (2.21)$$

Dove:

$$a_E = d_e \Delta y \quad (2.22)$$

$$a_w = d_w \Delta y \quad (2.23)$$

$$a_n = d_n \Delta x \quad (2.24)$$

$$a_s = d_s \Delta x \quad (2.25)$$

$$a_p = a_E + a_w + a_n + a_s \quad (2.26)$$

$$b^* = [u_e^* - u_w^*] \Delta y + [v_n^* - v_s^*] \Delta x \quad (2.27)$$

L'ultimo parametro **b*** costituisce una relazione di continuità discretizzata scritta per le componenti di velocità imperfette u^* e v^* : affinché tali componenti rispettino la continuità, b^* deve annullarsi, in questo modo non sarà più necessario correggere i campi di velocità. Questo è un aspetto chiave che verrà sfruttato come **criterio di convergenza** per l'algoritmo risolutivo utilizzato in Matlab. Di fatti la convergenza è ottenuta quando i valori di p' o del termine sorgente di massa b^* diventano minori di una certa tolleranza. Nelle simulazioni è stato applicato **sottorilassamento** su pressione e velocità per compensare nei primi cicli iterativi valori di p' troppo elevati, i quali potrebbero compromettere la convergenza.

Si è usata quindi una relazione del tipo:

$$p^{(n-1)} = p^{*(n-1)} + \alpha_p p'^{(n-1)} \quad (2.28)$$

Vieni mostrato in figura l'algoritmo SIMPLE implementato in Matlab:

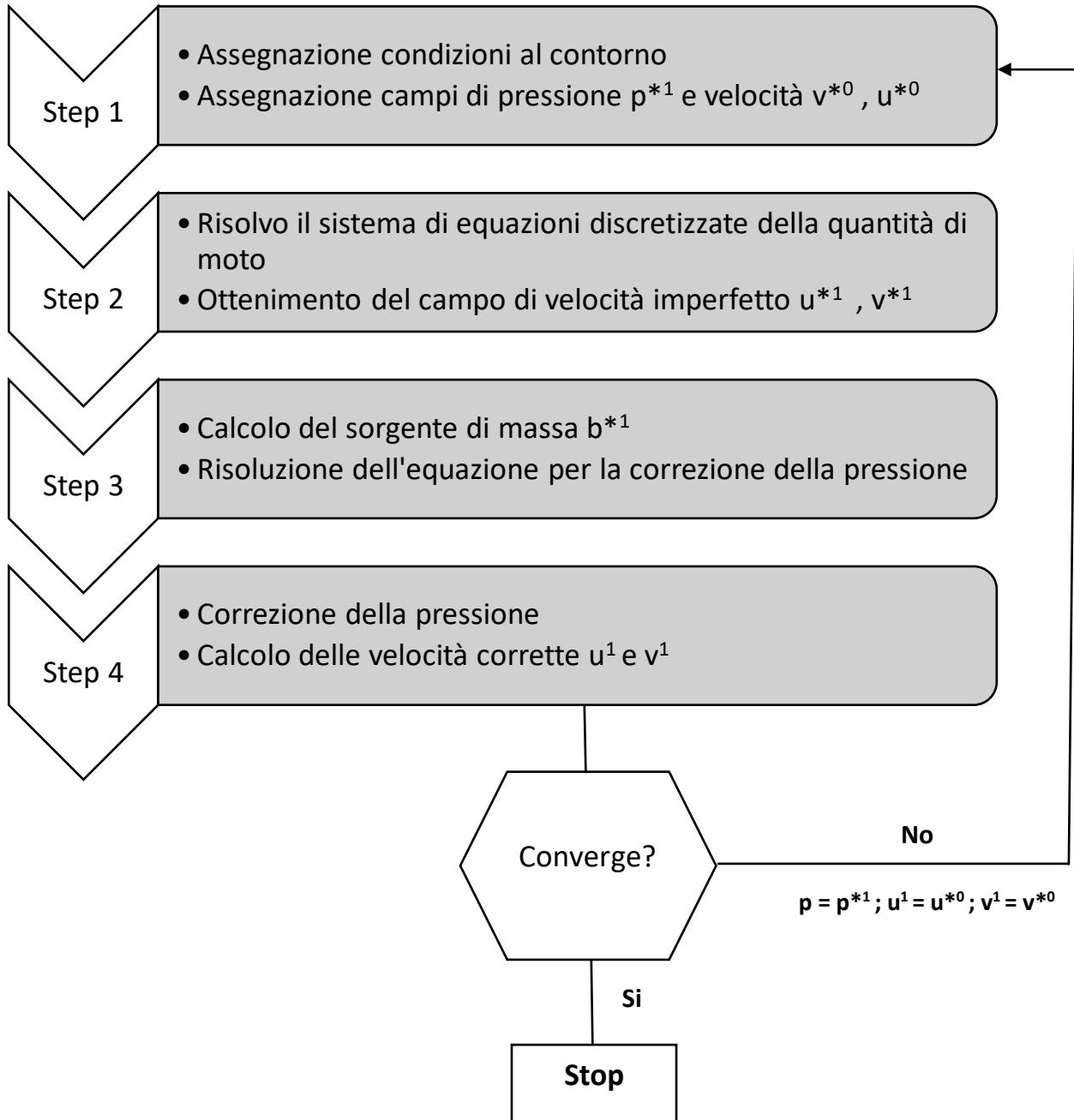


Figura 5: algoritmo SIMPLE

Come secondo metodo di risoluzione è stato utilizzato l'algoritmo **SIMPLER** (*SIMPLE-Revised*), il cui obiettivo è migliorare la velocità di convergenza del SIMPLE. La lentezza del SIMPLE è essenzialmente dovuta alla formulazione approssimata usata per p' , in particolare nella correzione della pressione. Con l'algoritmo SIMPLER si usa l'equazione per p' soltanto per la correzione delle velocità e si ricava poi una diversa equazione per la pressione corretta p . Dalla forma discretizzata dell'equazione per la conservazione della quantità di moto :

$$u_e = \frac{\sum_{nb} a_{nb} u_{nb} + b}{a_e} + d_e(p_p - p_e) \quad (2.29)$$

trascurando il termine di pressione si ricavano le “pseudo-componenti” di velocità:

$$\hat{u}_e = \frac{\sum_{nb} a_{nb} u_{nb} + b}{a_e} \quad (2.30)$$

Confrontando la 2.29 e 2.30:

$$u_e = \hat{u}_e + d_e(p_p - p_e) \quad (2.31)$$

$$v_n = \hat{u}_n + d_n(p_p - p_n) \quad (2.32)$$

A questo punto il campo di pressione p da usare nelle 2.31 e 2.32 è dato dalla soluzione della seguente equazione, formalmente identica alla 2.21 per la correzione di pressione p' :

$$a_p p_p = a_e p_e + a_w p_w + a_n p_n + a_s p_s - \hat{b} \quad (2.33)$$

con:

$$\hat{b} = (\hat{u}_e - \hat{u}_w)\Delta y + (\hat{v}_n - \hat{v}_s)\Delta x \quad (2.34)$$

In figura è riportato il diagramma di flusso dell'algoritmo SIMPLER:

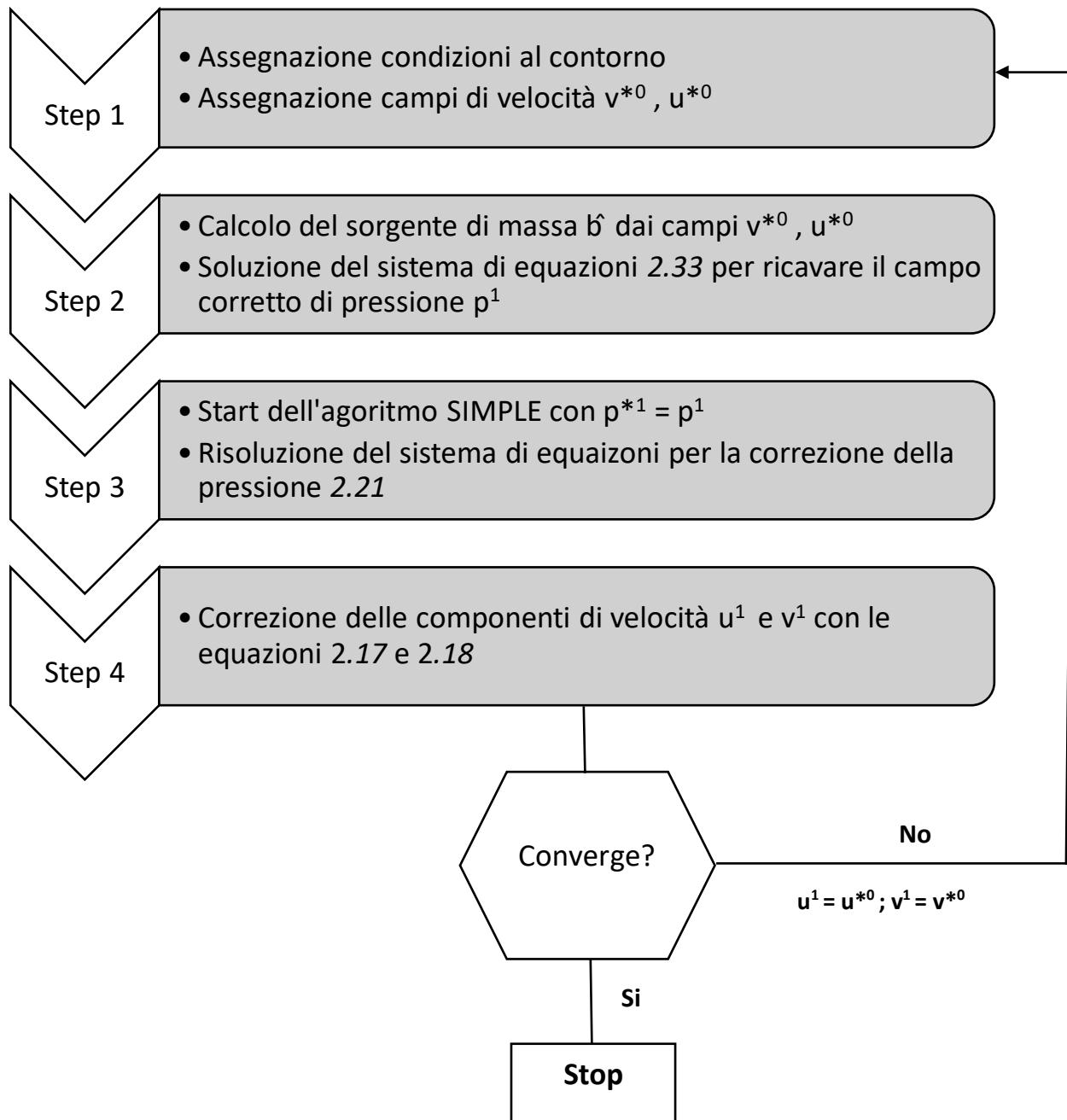


Figura 6: algoritmo SIMPLER

3. Risultati caso 1

In questo paragrafo vengono mostrati i risultati delle **simulazioni relative al Caso 1**, ovvero il canale dritto. Come già accennato, come condizioni di avvenuta **convergenza** dell'algoritmo sono stati monitorati gli andamenti dei massimi valori assunti sia da b^* che di p' . In aggiunta a questi parametri, è stato utile calcolare per b^* , soprattutto per l'implementazione del SIMPLER, lo scarto tra l' i -esima iterazione e la precedente, rapportando il tutto al valore di b^* relativo alla prima iterazione:

$$\Delta b^* = \frac{b^*(i) - b^*(i-1)}{b^*(i=1)} \quad (3.1)$$

Come ulteriore verifica e conferma dell'avvenuta convergenza è stata calcolata per ogni iterazione la **portata** in corrispondenza dell'outlet: di fatto è comune osservare l'assestamento della portata ben prima che i residui abbiano raggiunto la tolleranza scelta. In definitiva verranno mostrati:

- a. Andamento dei residui b^* e p' ;
- b. Contour plot del campo di velocità u ;
- c. Contour plot del campo di velocità v ;
- d. Contour plot del campo di pressione p (statica, dinamica e totale);
- e. Andamento della **portata** all'outlet ;
- f. Confronto tra **soluzione numerica stazionaria** e **soluzione analitica** con il profilo teorico di Poiseuille ricavato dall'equazione di *Hagen-Poiseuille*:

$$u = \frac{\Delta p}{2\mu L} y(H - y) \quad (3.2)$$

dove Δp è il salto di pressione tra inlet e outlet, esprimibile in funzione della velocità in ingresso u_{inlet} , delle proprietà del fluido e della geometria del problema (con A sezione trasversale del canale):

$$\Delta p = u_{in} A \frac{12\mu L}{H^3} \quad (3.3)$$

Le tolleranze associate ai residui sono le stesse per Matlab e per Fluent, così come i valori di sottorilassamento e la mesh usata:

Tolleranza b^*	$1 * 10^{-8}$
Tolleranza p'	$1 * 10^{-8}$
α_p	0.3
α_u	0.7
α_v	0.7

Tabella 5: tolleranze e sottorilassamenti per il caso 1

I. SIMPLE – upwind

Re	n	m
50	60	20

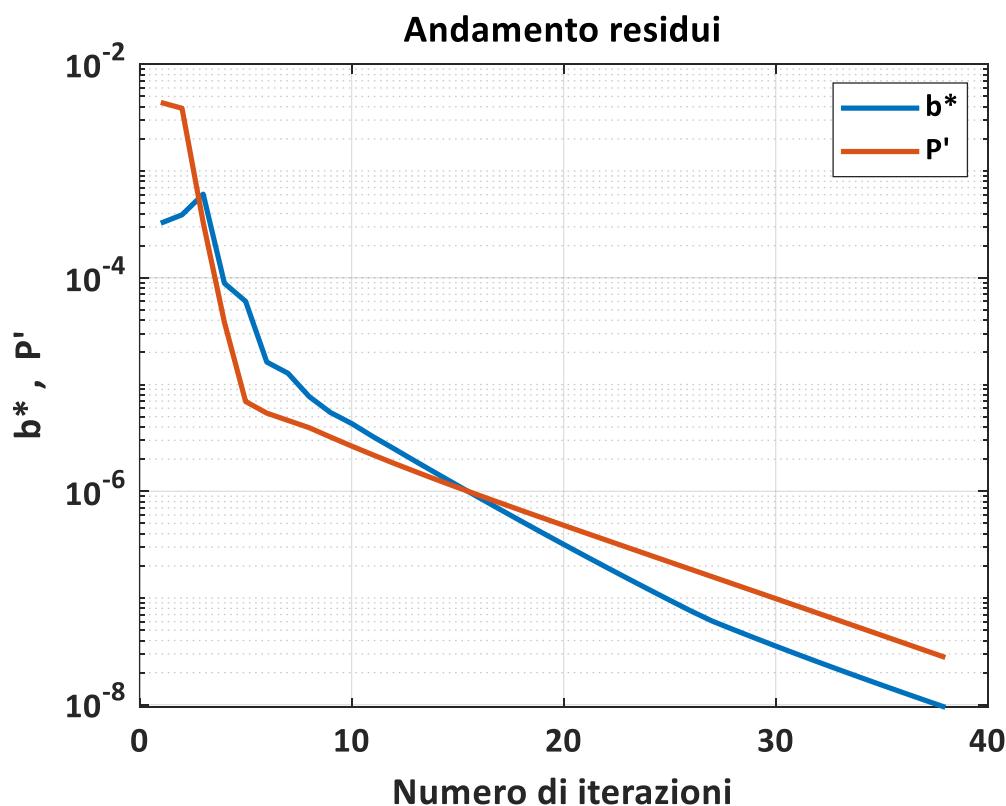


Figura 6: residui SIMPLE-upwind, Re=50 - mesh 60x20

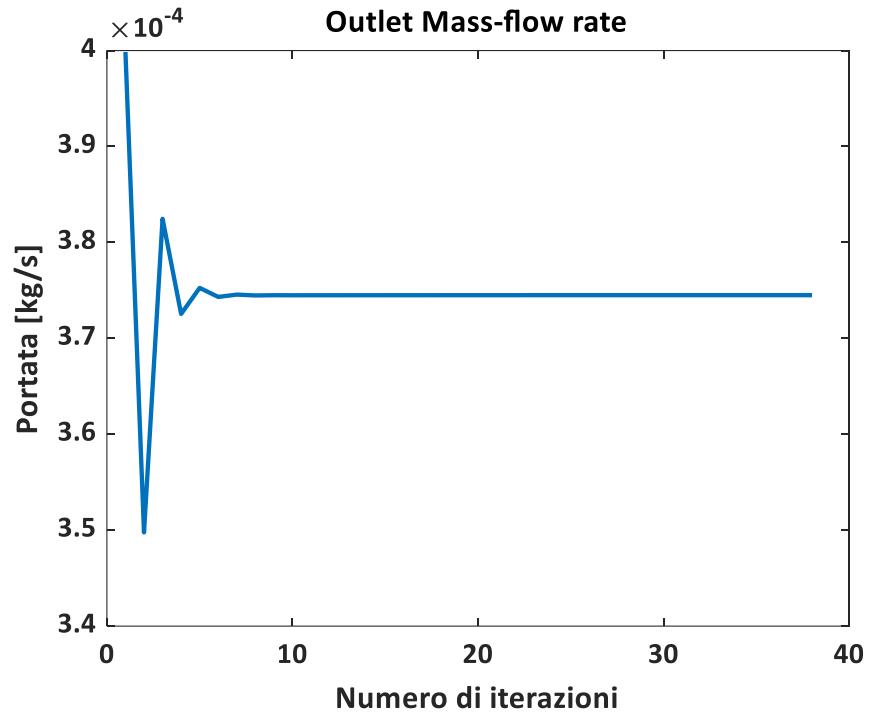


Figura 7: andamento portata outlet, $Re=50$ - mesh 60×20

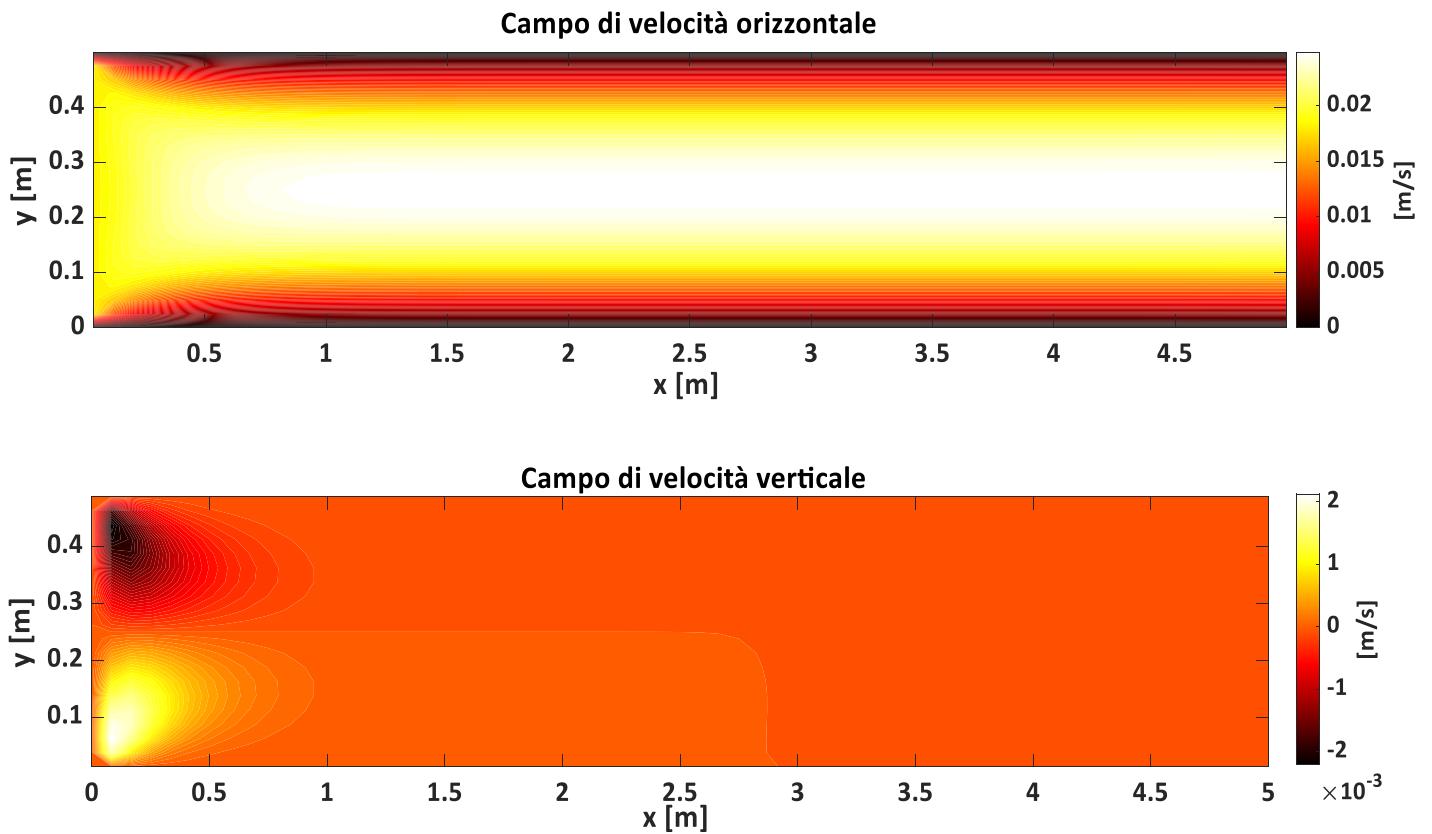


Figura 8: contour plot campi di velocità, $Re=50$ - mesh 60×20

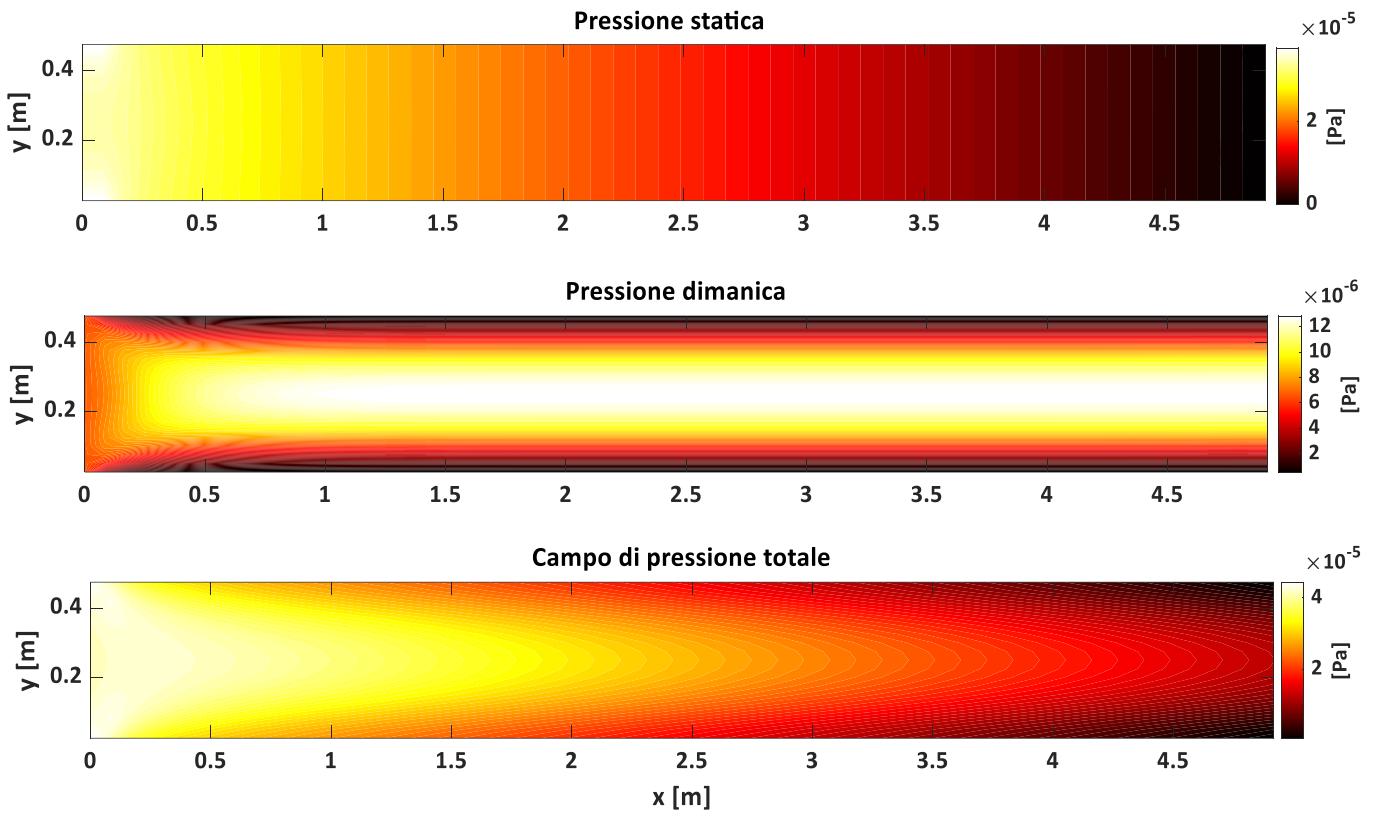


Figura 9: contour plot campi di pressione, $Re=50$ - mesh 60x20

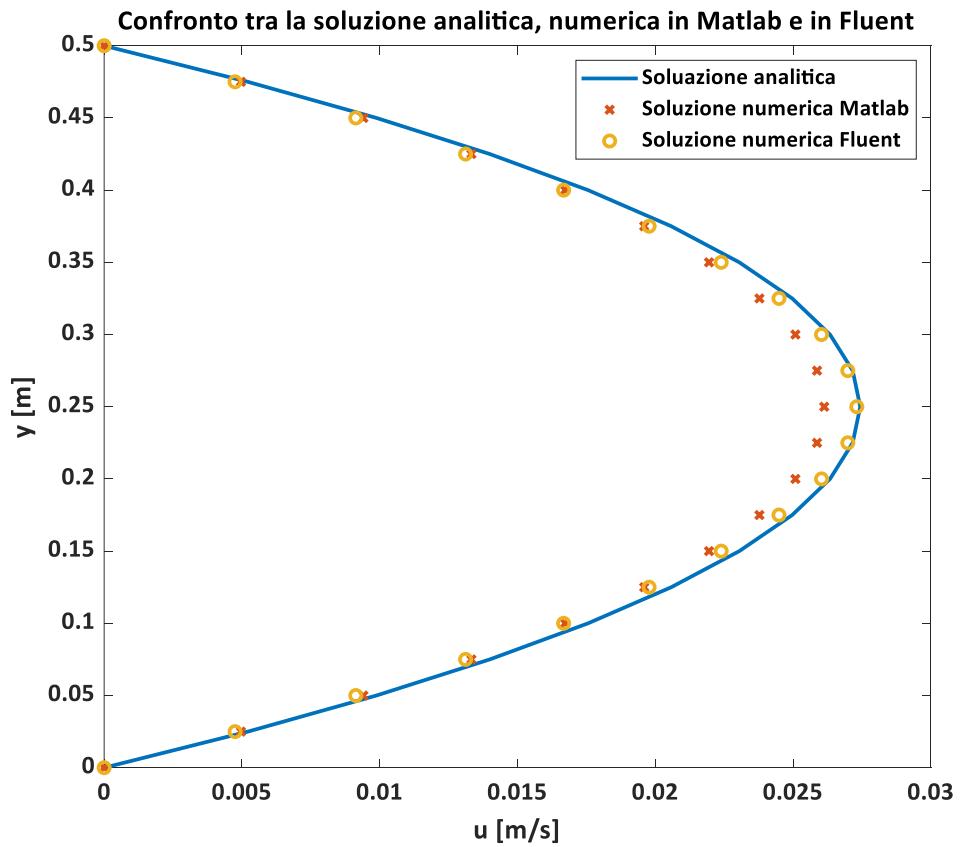


Figura 10: confronto tra soluzione numerica e analitica, $Re=50$ - mesh 60x20

Vengono riassunti ora i principali risultati delle simulazioni in termini di numero di iterazioni e deviazione standard:

$$\sigma = \sqrt{\frac{\sum_{k=1}^N (u_k^{(\text{num})} - u_k^{(\text{Poiseuille})})^2}{N}} \quad (3.3)$$

Reynolds	Mesh	Iterazioni	σ_{matlab}	σ_{Fluent}
Re = 50	60x20	38	$5.5917 \cdot 10^{-4}$	$3.5719 \cdot 10^{-4}$
	120x40	49	$5.1991 \cdot 10^{-4}$	$9.1052 \cdot 10^{-6}$
Re = 100	60x20	49	$1.1 \cdot 10^{-3}$	$7.1438 \cdot 10^{-4}$
	120x40	63	$9.0735 \cdot 10^{-4}$	$2.1502 \cdot 10^{-5}$

Tabella 6: risultati SIMPLE-upwind caso 1

Le deviazioni standard ottenute con Matlab rimangono confrontabili tra loro al variare della mesh e del Re sebbene diminuiscano all'infittirsi della griglia, mentre in Fluent si evidenzia una diminuzione significativa di σ passando dalla prima alla seconda griglia.

II. SIMPLE – differenze centrate

Re	n	m
50	60	20

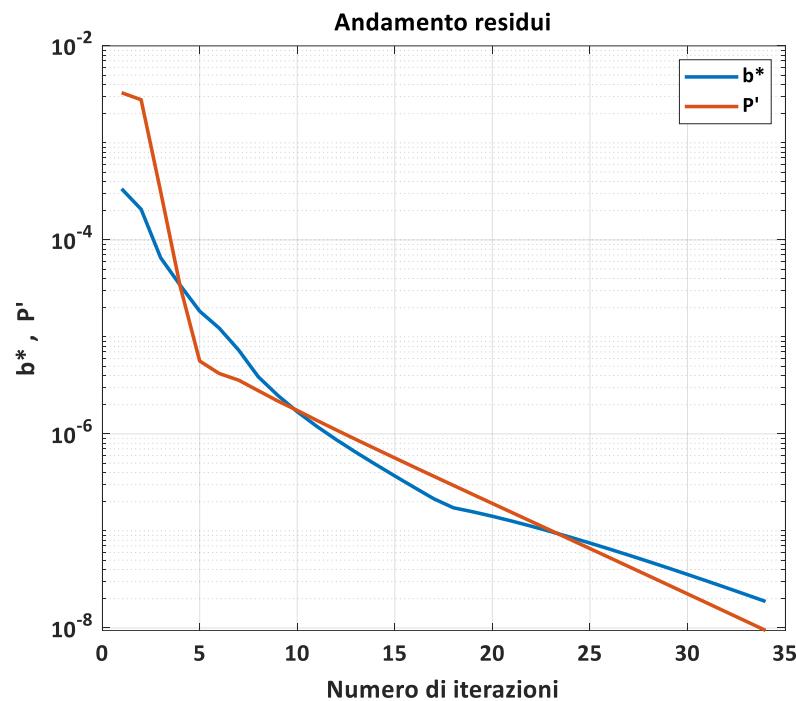


Figura 11: residui SIMPLE-differenze centrate, $Re=50$ - mesh 60×20

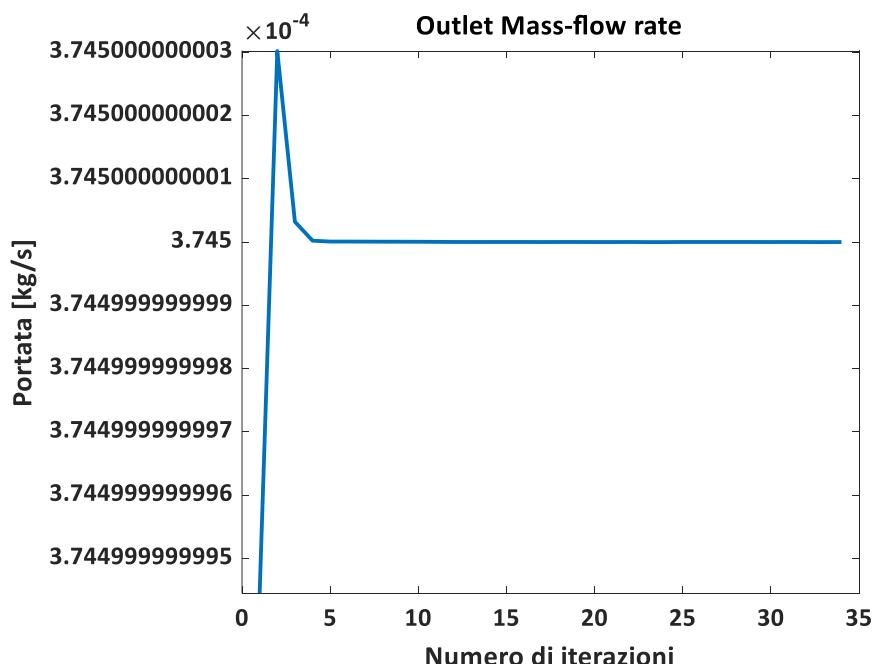


Figura 12: andamento portata outlet, $Re=50$ - mesh 60×20

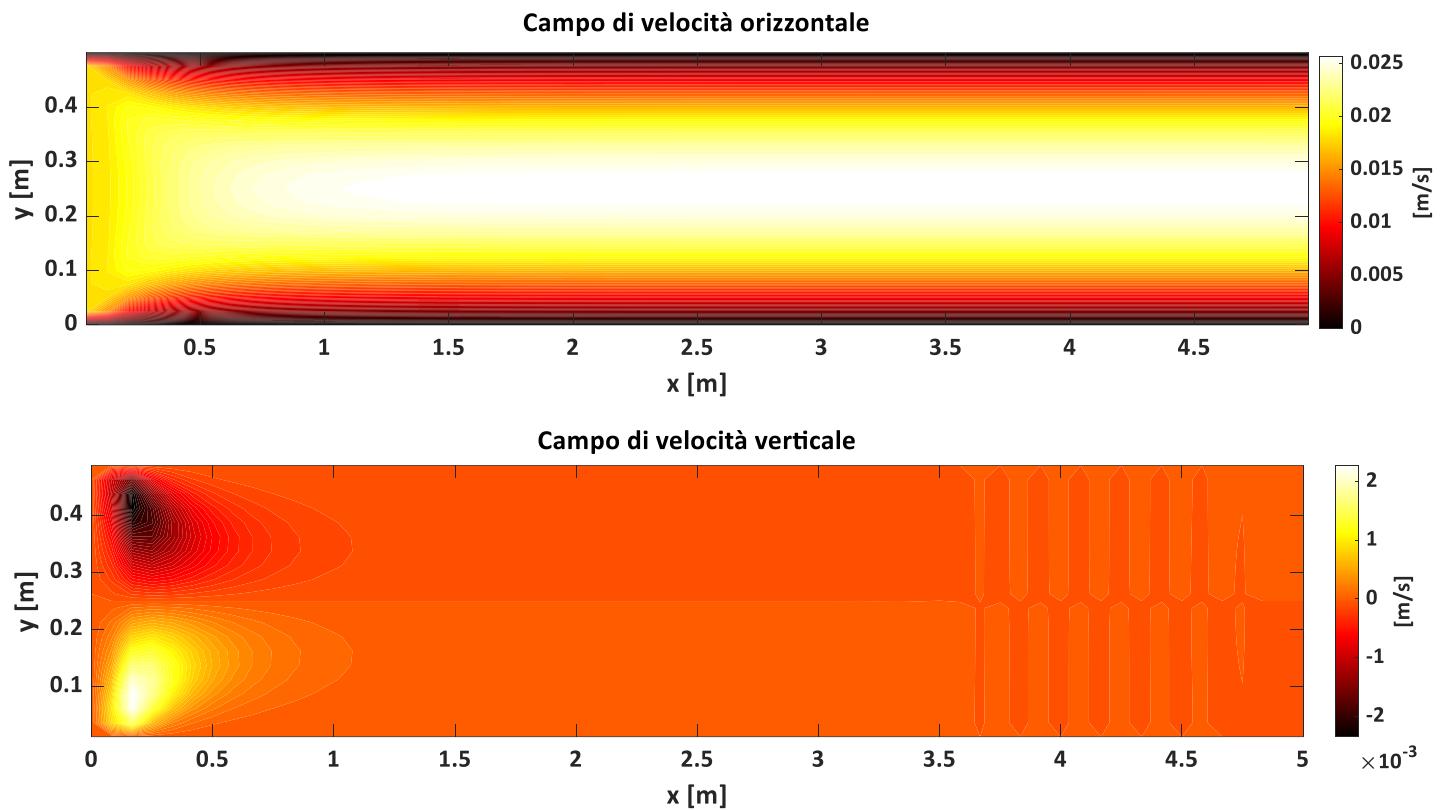


Figura 13: contour plot campi di velocità, $Re=50$ - mesh 60×20

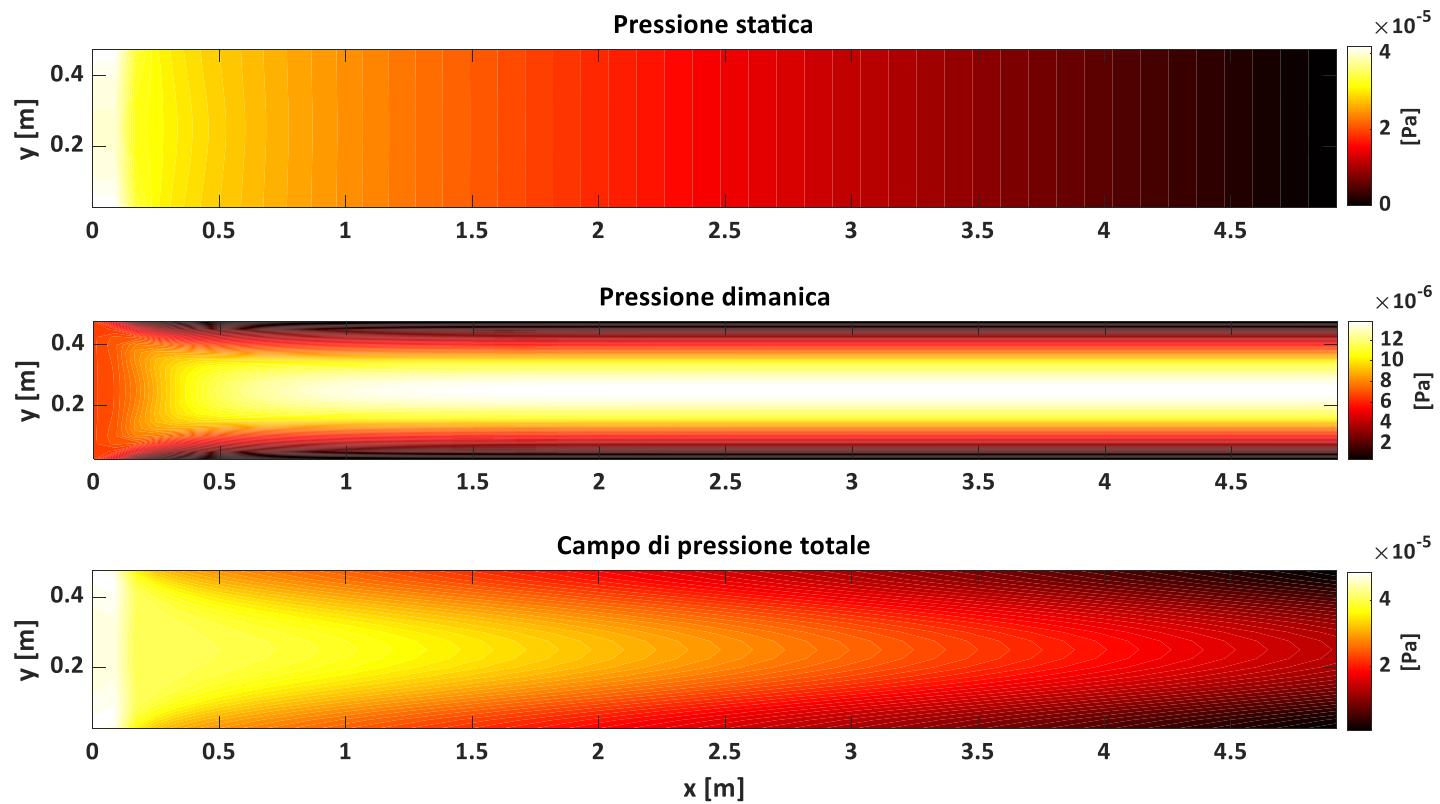


Figura 14: contour plot campi di pressione, $Re=50$ - mesh 60×20

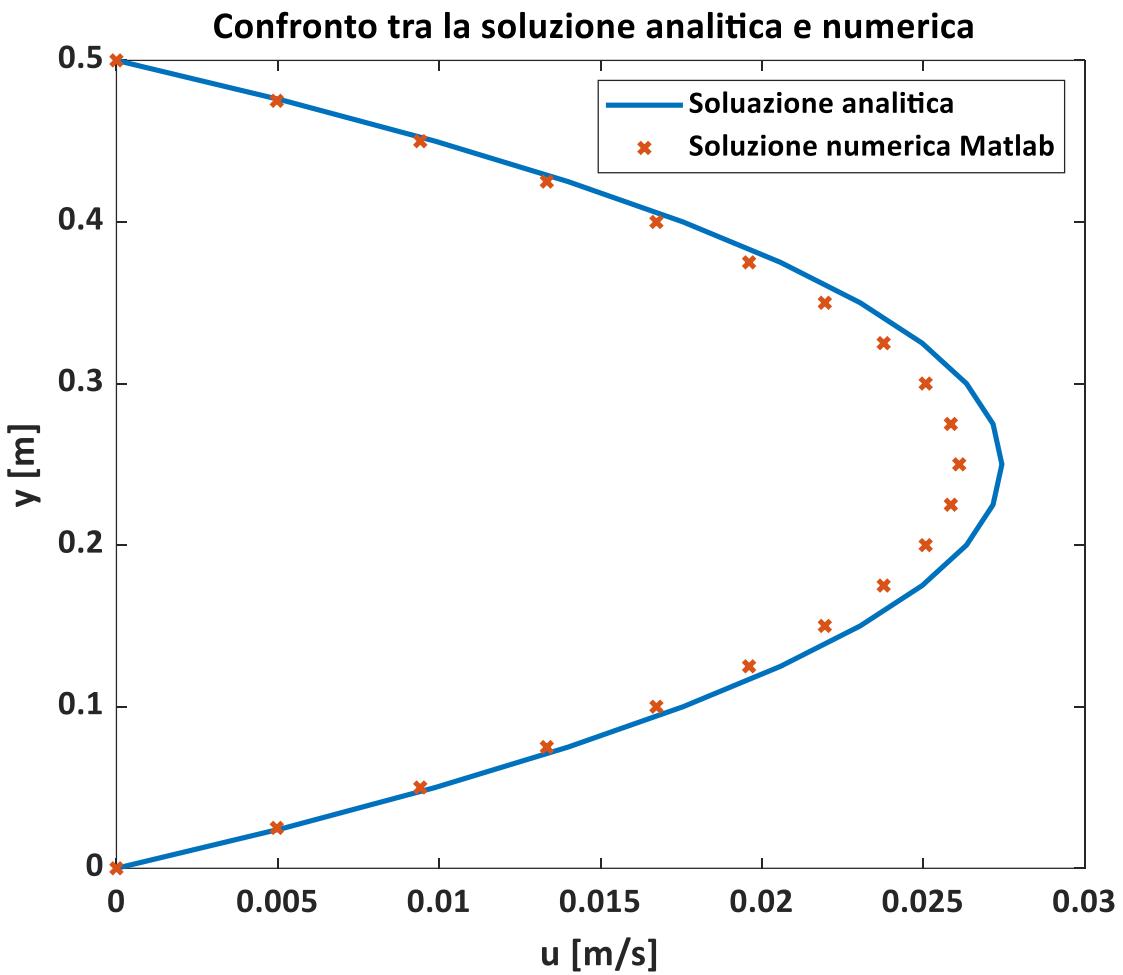


Figura 13: confronto tra soluzione analitica e numerica, $Re=50$ - mesh 60×20

Reynolds	Mesh	Iterazioni	σ_{matlab}
Re = 50	60x20	48	$5.5689 \cdot 10^{-4}$
	120x40	45	$3.0639 \cdot 10^{-4}$
Re = 100	60x20	96	$1.1 \cdot 10^{-3}$
	120x40	51	$5.9127 \cdot 10^{-4}$

Tabella 7: risultati SIMPLE-differenze centrate caso 1

III. SIMPLER – upwind

Re	n	m
50	60	20

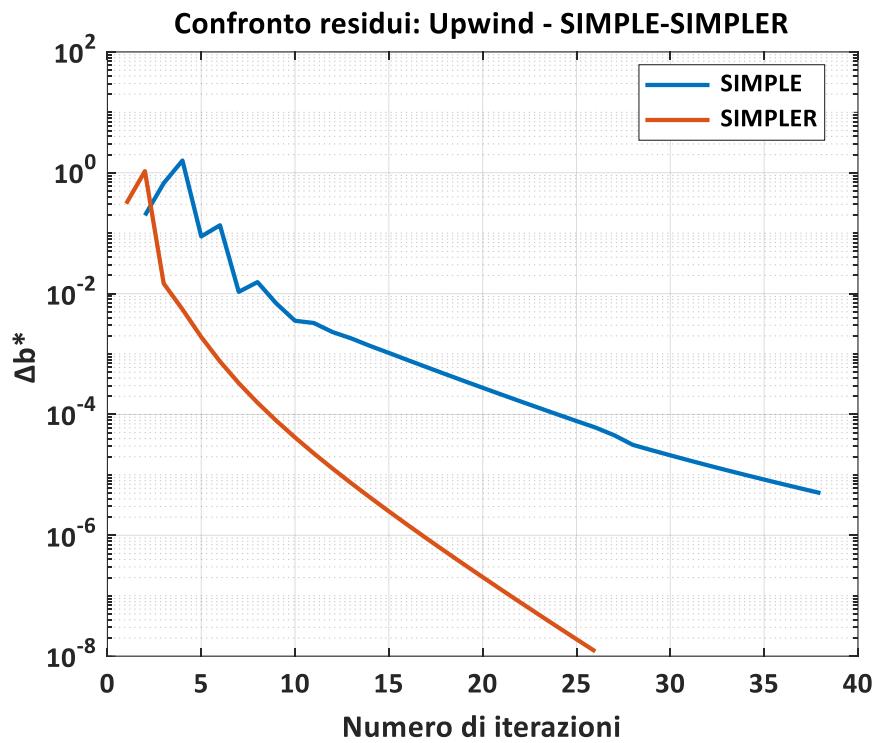


Figura 14: confronto Δb^* SIMPLE E SIMPLER - upwind, $Re=50$ - mesh 60×20

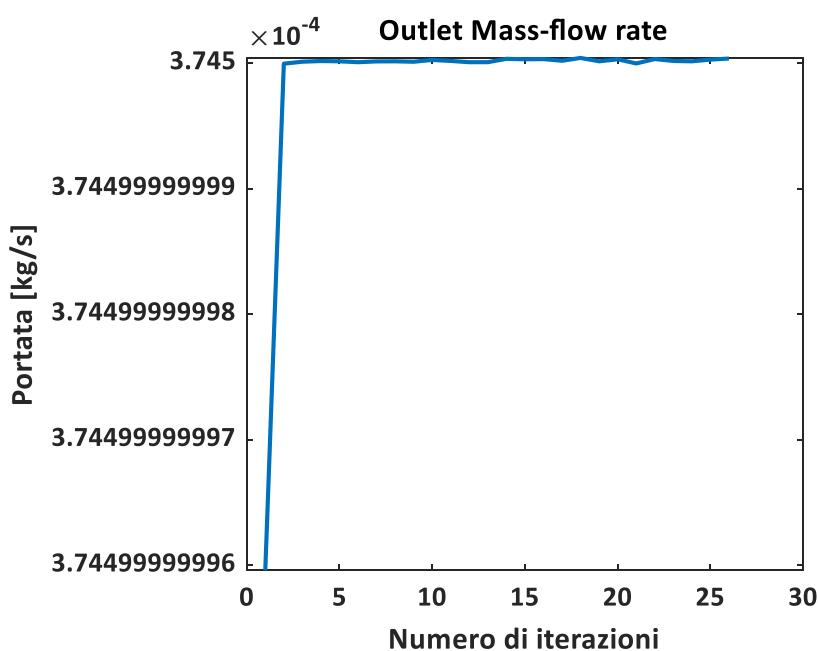


Figura 15: andamento portata outlet, $Re=50$ - mesh 60×20

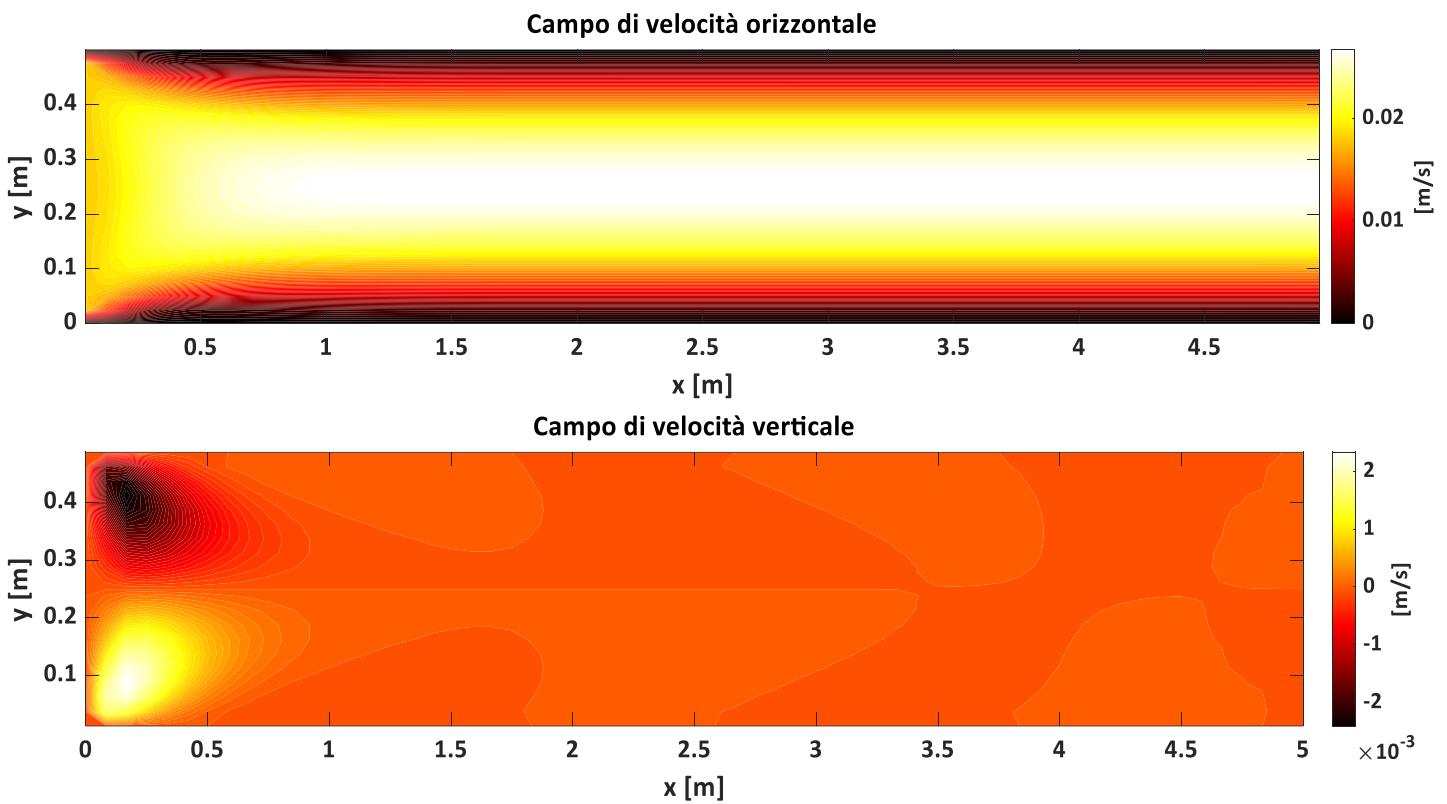


Figura 18: contour plot campi di velocità, $Re=50$ - mesh 60×20

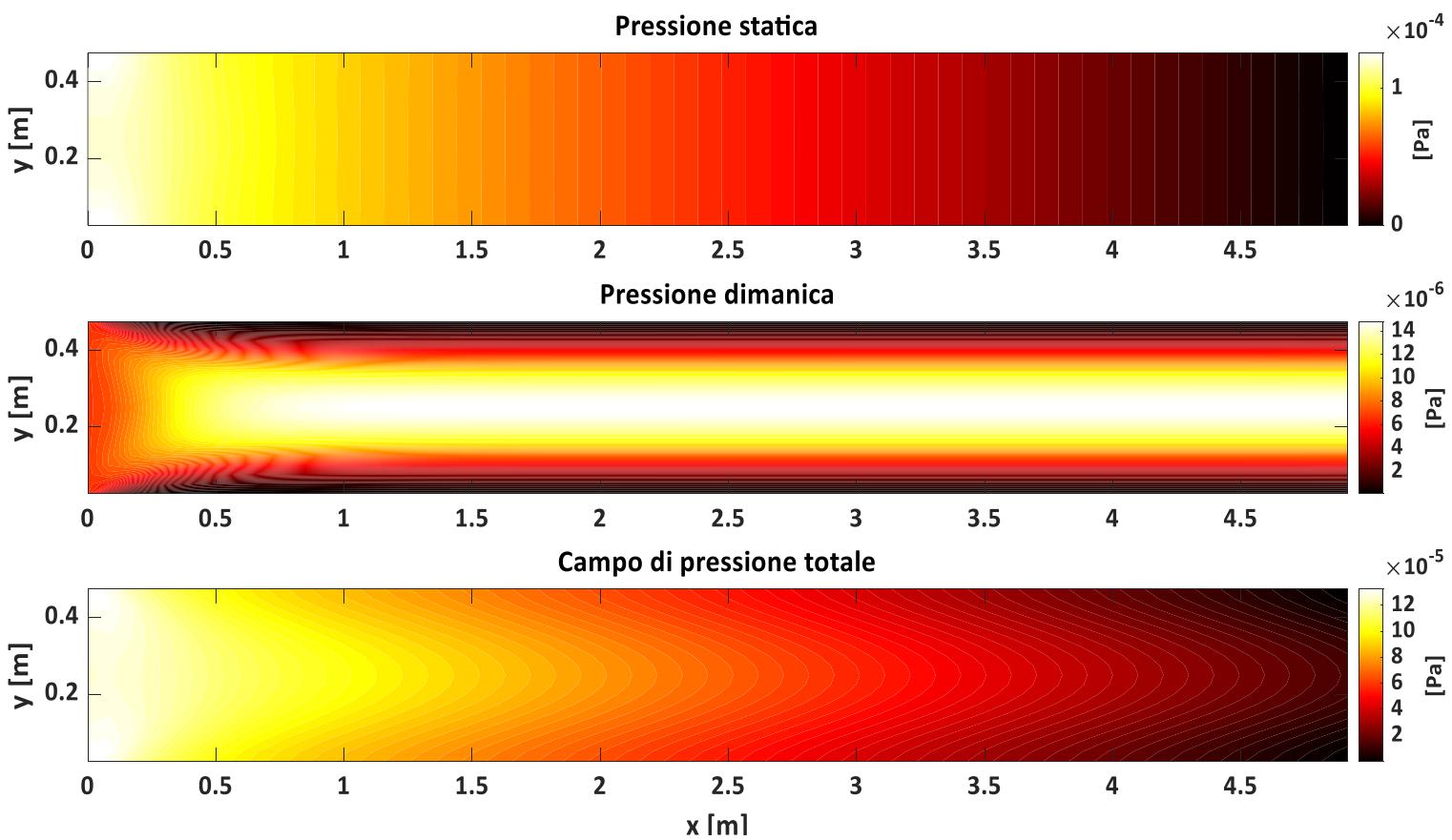


Figura 19: contour plot campi di pressione, $Re=50$ - mesh 60×20

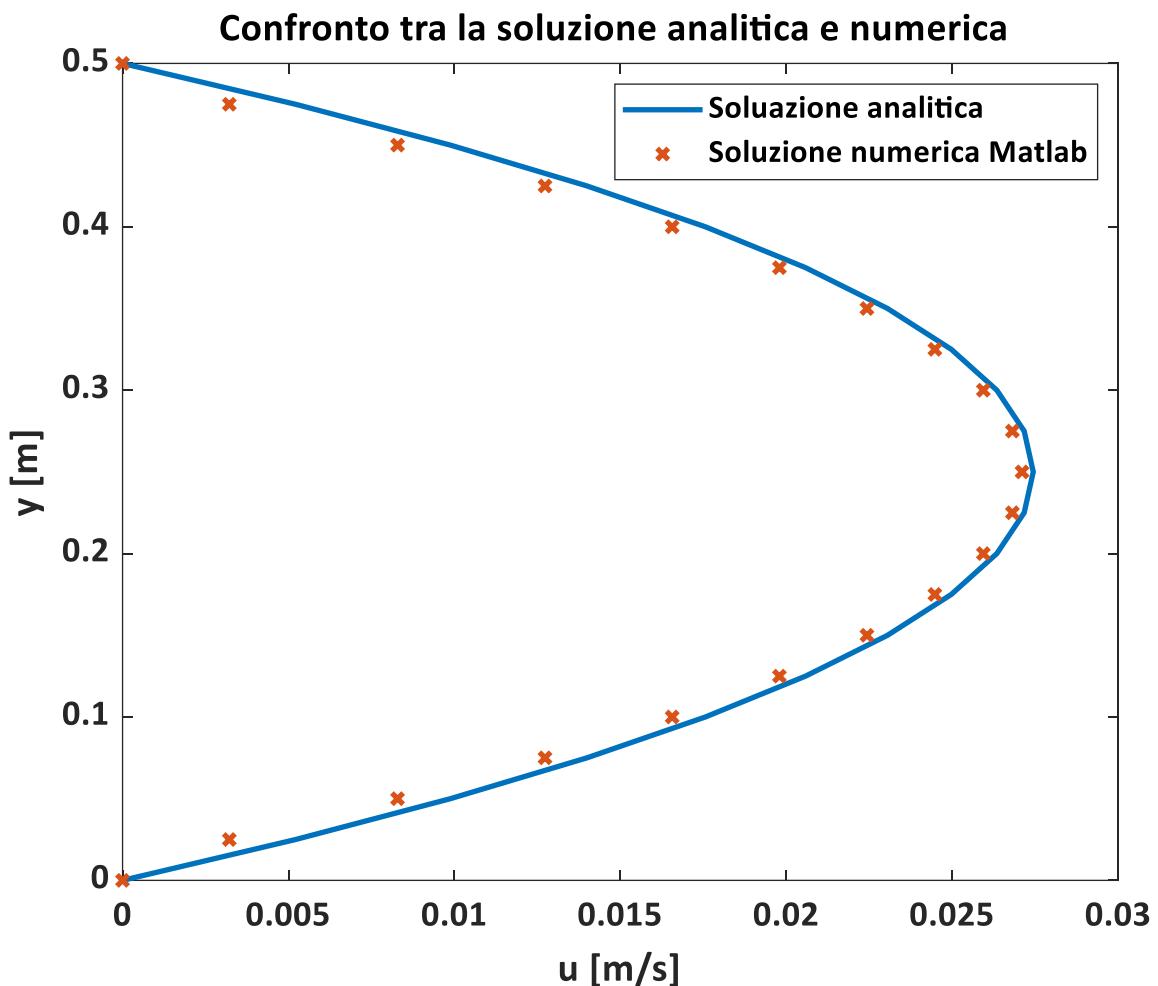


Figura 16: confronto tra soluzione analitica e numerica, $Re=50$ - mesh 60x20

Reynolds	Mesh	Iterazioni SIMPLER	σ_{SIMPLER}	Iterazioni SIMPLE	σ_{SIMPLE}
Re = 50	60x20	11	$5.8163 \cdot 10^{-4}$	38	$5.5917 \cdot 10^{-4}$
	120x40	33	$3.0609 \cdot 10^{-4}$	49	$5.1991 \cdot 10^{-4}$
Re = 100	60x20	19	$1.1 \cdot 10^{-3}$	49	$1.1 \cdot 10^{-3}$
	120x40	31	$7.8822 \cdot 10^{-4}$	63	$9.0735 \cdot 10^{-4}$

Tabella 8: confronto risultati SIMPLER-SIMPLE, upwind - caso 1

IV. SIMPLER – differenze centrate

Re	n	m
50	60	20

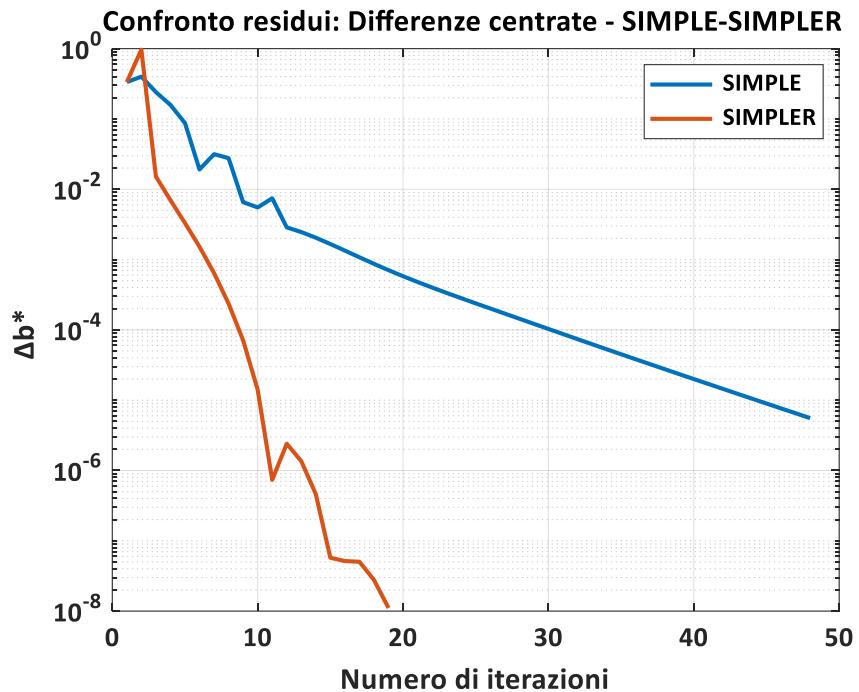


Figura 17: confronto Δb^* SIMPLE-SIMPLER, differenze centrate, $Re=50$ - mesh 60×20

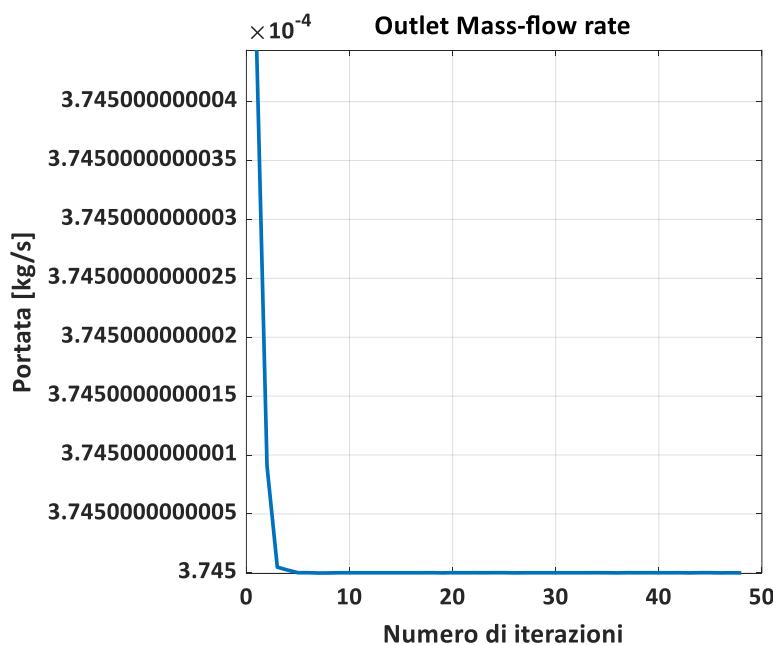
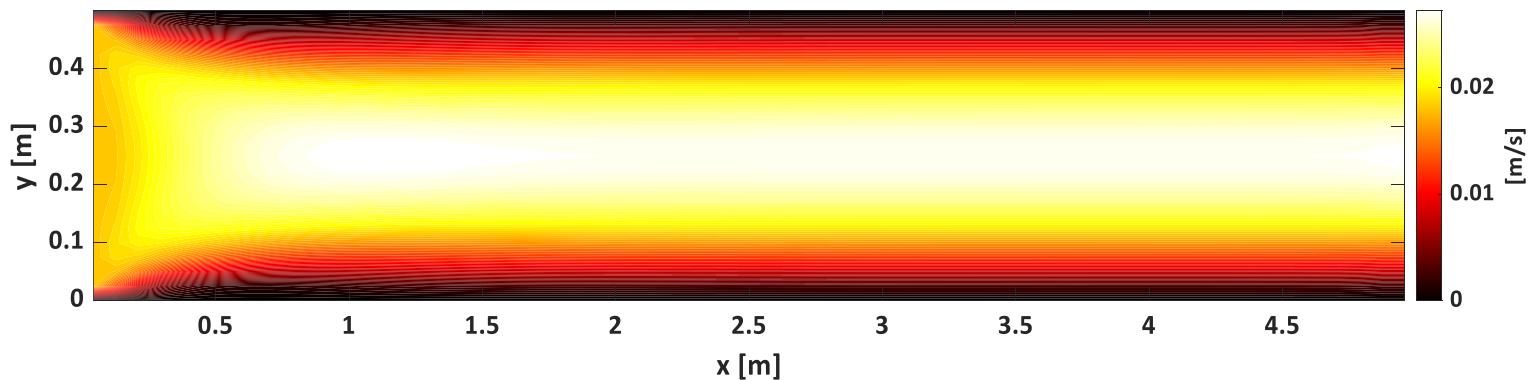


Figura 18: andamento portata outlet, $Re=50$ - mesh 60×20

Campo di velocità orizzontale



Campo di velocità verticale

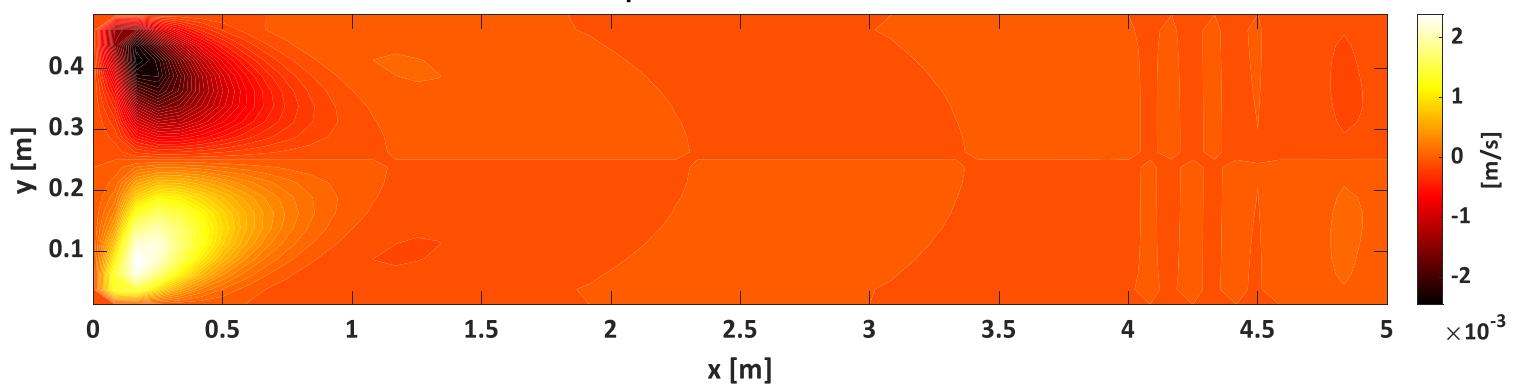
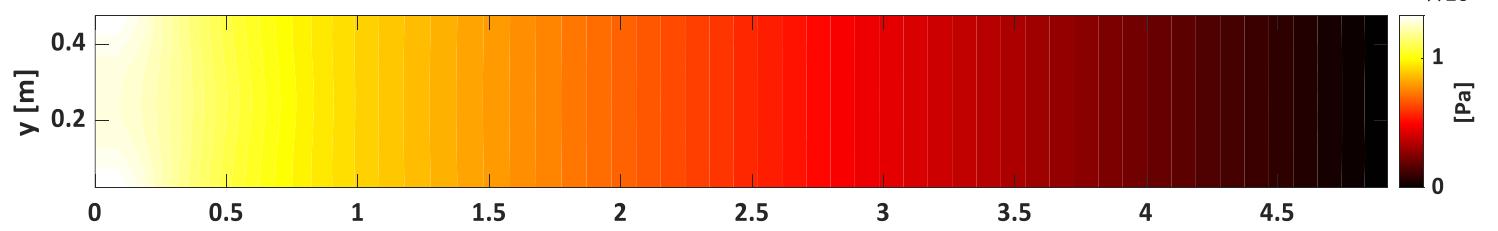
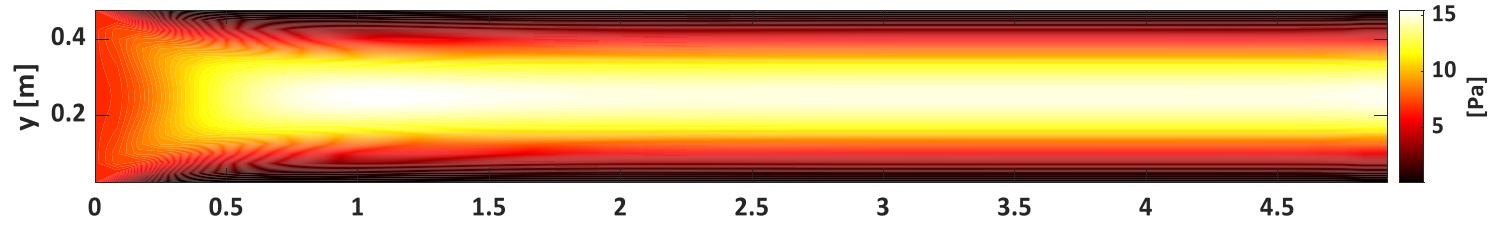


Figura 23: contour plot campi di velocità , $Re=50$ - mesh 60×20

Pressione statica



Pressione dinamica



Campo di pressione totale

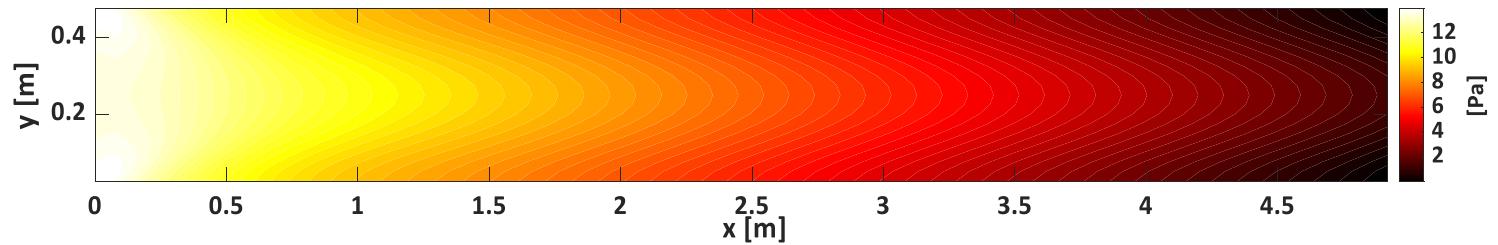


Figura 24: contour plot campi di pressione, $Re=50$ - mesh 60×20

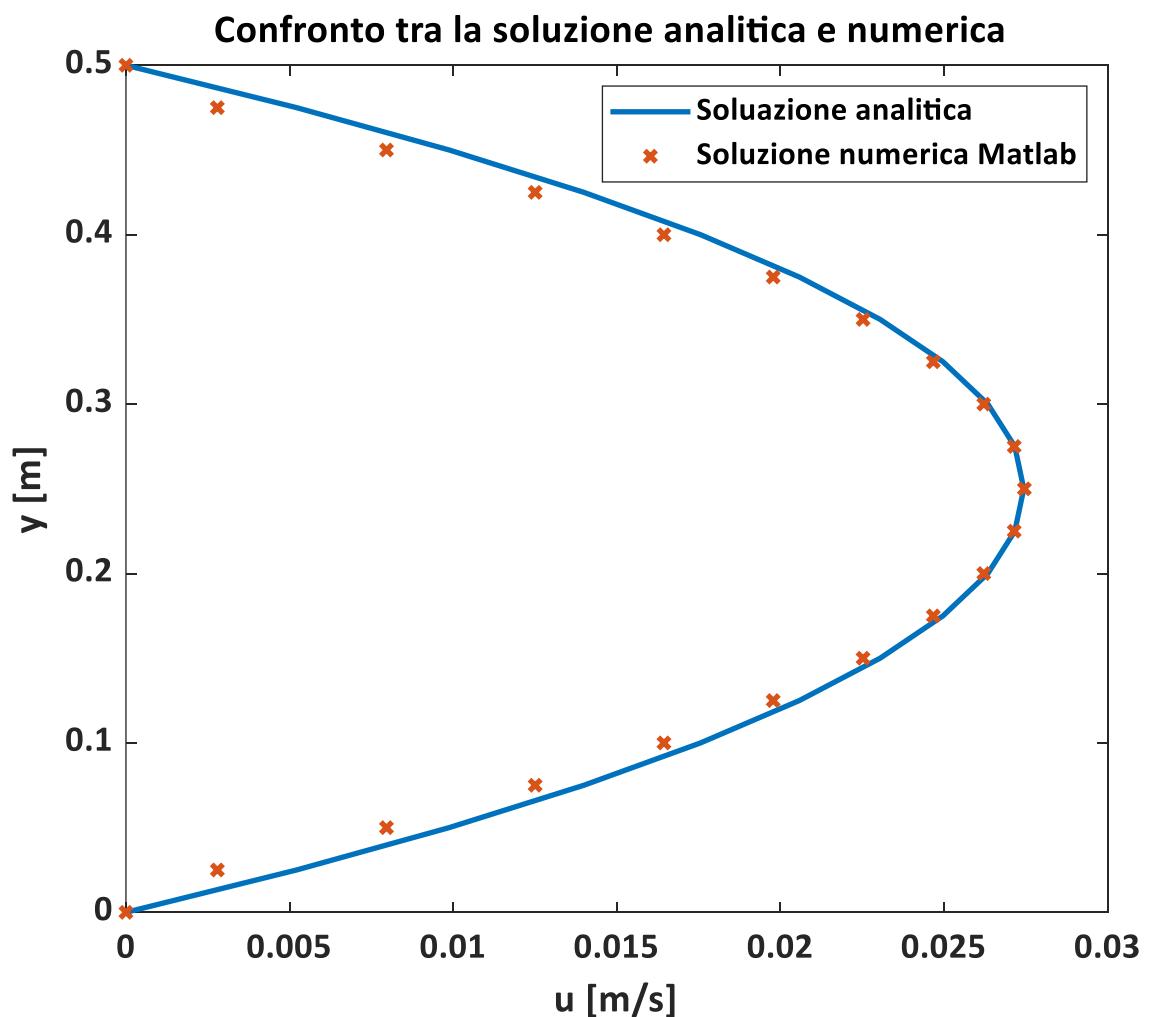


Figura 19: confronto soluzione analitica e numerica , $Re=50$ - mesh 60×20

Reynolds	Mesh	Iterazioni SIMPLER	σ_{simpler}	Iterazioni SIMPLE	σ_{simple}
Re = 50	60x20	19	$6.8893 \cdot 10^{-4}$	48	$5.5689 \cdot 10^{-4}$
	120x40	35	$2.7058 \cdot 10^{-4}$	45	$3.0639 \cdot 10^{-4}$
Re = 100	60x20	14	$1.5 \cdot 10^{-3}$	96	$1.1 \cdot 10^{-3}$
	120x40	33	$5.4761 \cdot 10^{-4}$	51	$5.9127 \cdot 10^{-4}$

Tabella 9: confronto risultati SIMPLER-SIMPLE, differenze centrate

In tabella è presente un riepilogo di tutti i risultati ottenuti per il *caso 1*:

Schema	Reynolds	Mesh	Iterazioni SIMPLER	σ_{simpler}	Iterazioni SIMPLE	σ_{simple}
Upwind	Re = 50	60x20	11	$5.8163 \cdot 10^{-4}$	38	$5.5917 \cdot 10^{-4}$
		120x40	33	$3.0609 \cdot 10^{-4}$	49	$5.1991 \cdot 10^{-4}$
	Re = 100	60x20	19	$1.1 \cdot 10^{-3}$	49	$1.1 \cdot 10^{-3}$
		120x40	31	$7.8822 \cdot 10^{-4}$	63	$9.0735 \cdot 10^{-4}$
Differenze centrate	Re = 50	60x20	19	$6.8893 \cdot 10^{-4}$	48	$5.5689 \cdot 10^{-4}$
		120x40	35	$2.7058 \cdot 10^{-4}$	45	$3.0639 \cdot 10^{-4}$
	Re = 100	60x20	14	$1.5 \cdot 10^{-3}$	96	$1.1 \cdot 10^{-3}$
		120x40	33	$5.4761 \cdot 10^{-4}$	51	$5.9127 \cdot 10^{-4}$

Tabella 10: riepilogo risultati caso 1

Risulta evidente come a parità di Re e mesh, il SIMPLER vada a convergenza con un numero inferiore di iterazioni (oltreché con un risparmio notevole di tempo computazionale impiegato dalla CPU).

4. Risultati caso 2

Per questa simulazione, la cui geometria è stata illustrata nel *paragrafo 1*, continuano a valere i richiami e le considerazioni teoriche del *paragrafo 2*. Per la particolare geometria, al fine di calcolare la lunghezza di riattacco della vena fluida, è necessario tuttavia trasportare e imporre alcune delle condizioni al contorno, su velocità e pressione, al bordo del gradino. Per implementare gli algoritmi risolutivi di interesse, il caso è stato trattato utilizzando una griglia equivalente al *caso 1*, “spegnendo” opportunamente le celle di calcolo che precedono il gradino. La condizione di ricircolo che si presenta in prossimità del gradino viene valutata attraverso la grandezza x_r/H , con x_r pari alla lunghezza di riattacco del fluido: in questo modo è possibile confrontare i valori ottenuti in **Matlab** con quelli sperimentali riportati nel *paragrafo 1* e con quelli ottenuti in ambiente **Fluent** (per lo schema upwind). In particolare, tale grandezza è stata calcolata determinando, all’interno della matrice di calcolo, la posizione in cui la componente orizzontale di velocità u cambia di segno a valle del gradino (da negativa a positiva), condizione che equivale a verificare che lo sforzo di taglio a parete sia pari a zero. Per confrontare le lunghezze di riattacco ottenute in Matlab/Fluent con i valori sperimentali, si è usato il seguente parametro:

$$\Delta x_r = \frac{x_{numerica} - x_{sperimentale}}{x_{sperimentale}} \quad (4.1)$$

Tra i risultati vengono mostrati:

- Plot andamento b^* al variare del Re per SIMPLE e SIMPLER (schema upwind);
- Contour plot campi di velocità u e v ;
- Vector plot (Fluent) del vettore velocità con visualizzazione della condizione di ricircolo ($Re=50$ - mesh 60x20);
- Contour plot campo di pressione (statica, dinamica e totale) ottenuto in Matlab e in Fluent;
- Andamento della portata all’outlet;
- Tabella riassuntiva di tutti i risultati ottenuti al variare della mesh, Reynolds e schema di interpolazione.

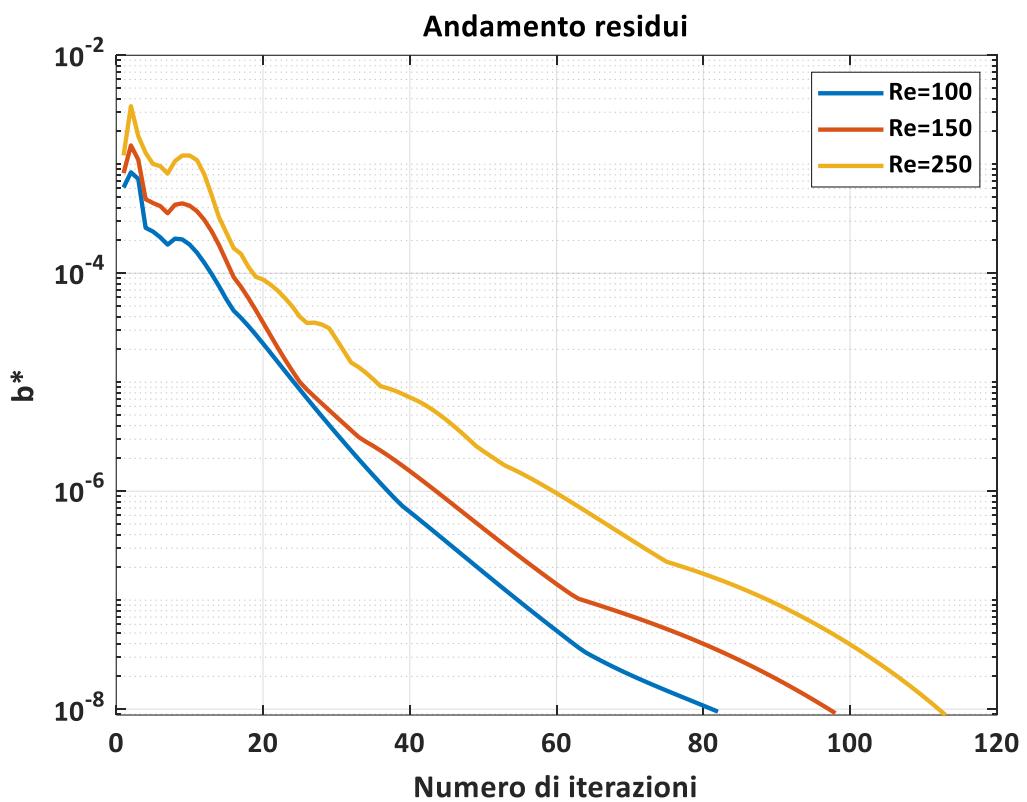


Figura 20: andamento b^* SIMPLE-upwind , $Re=[100-150-250]$ - mesh 60x20

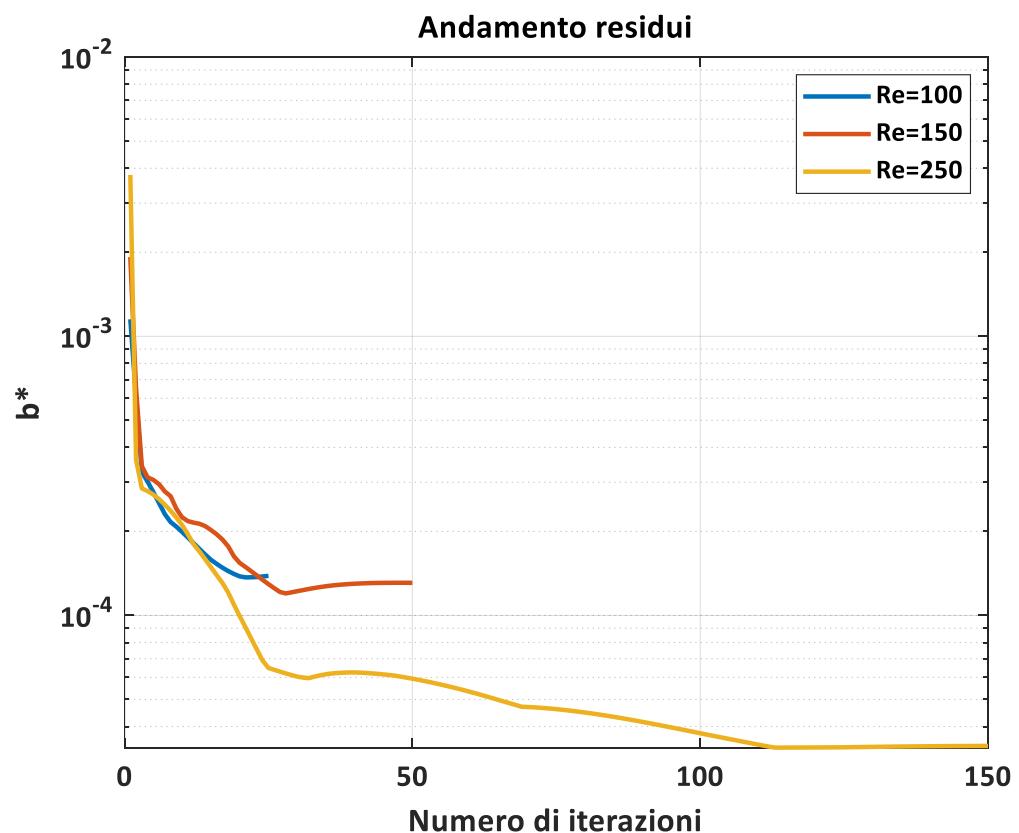


Figura 21: andamento b^* SIMPLER-upwind , $Re=[100-150-250]$ - mesh 60x20

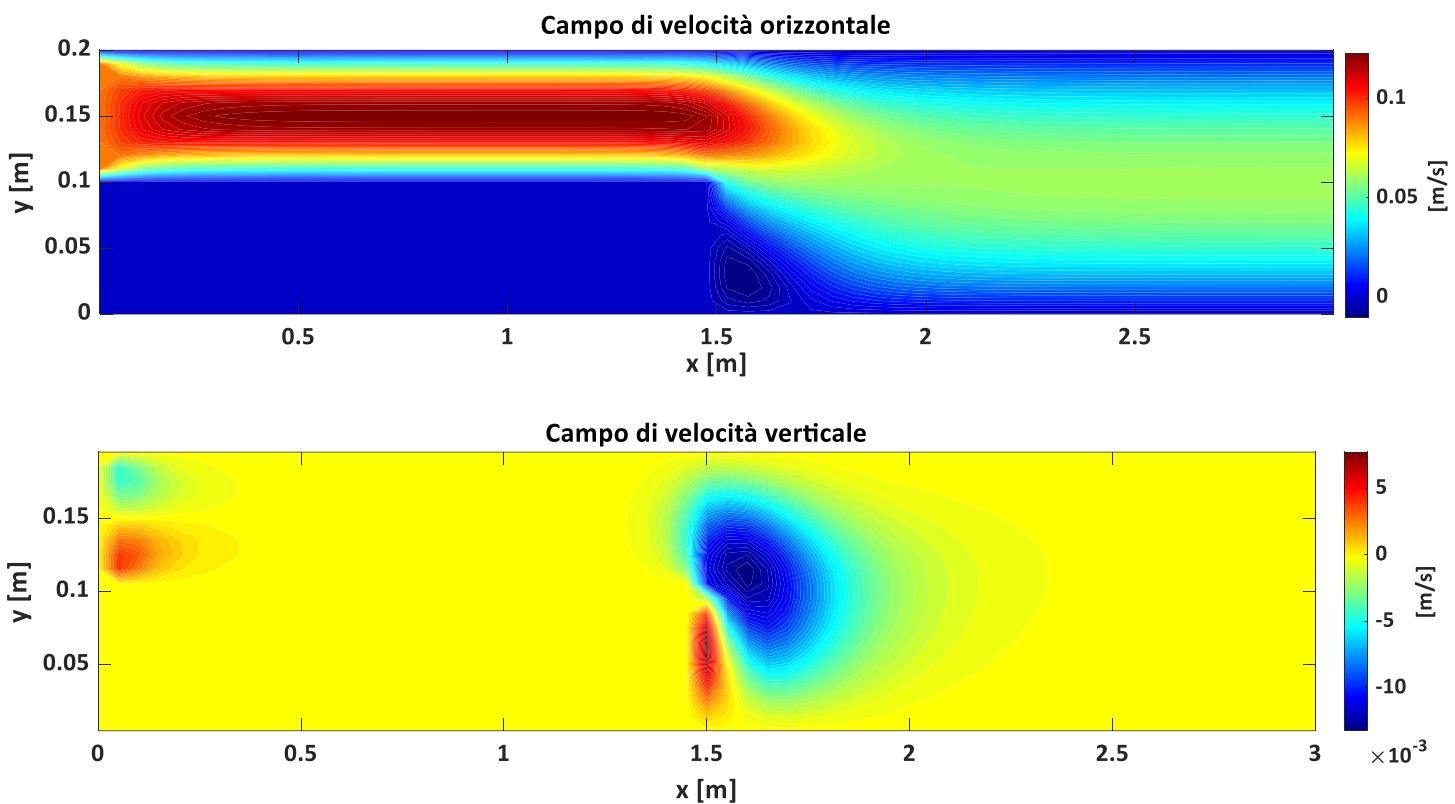


Figura 28: contour plot campi di velocità , $Re=100$ - mesh 60×20

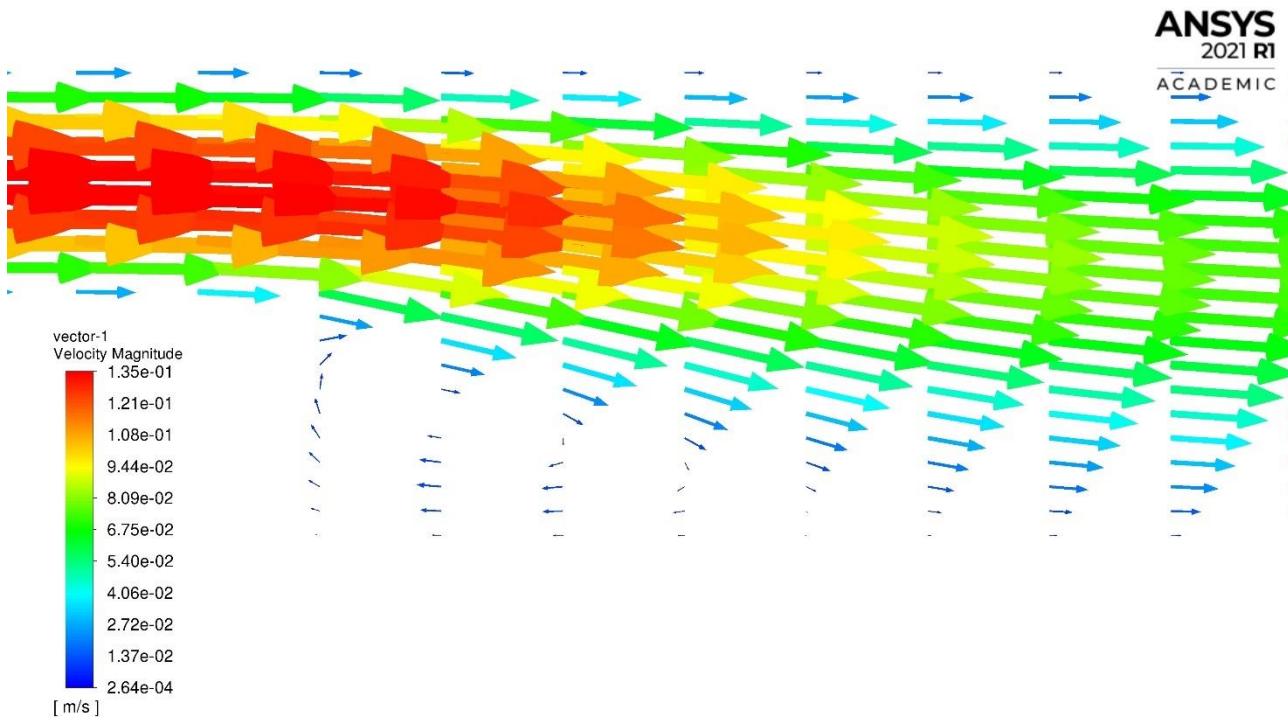


Figura 29: condizione di ricircolo: vector plot campo di velocità Fluent , $Re=100$ - mesh 60×20

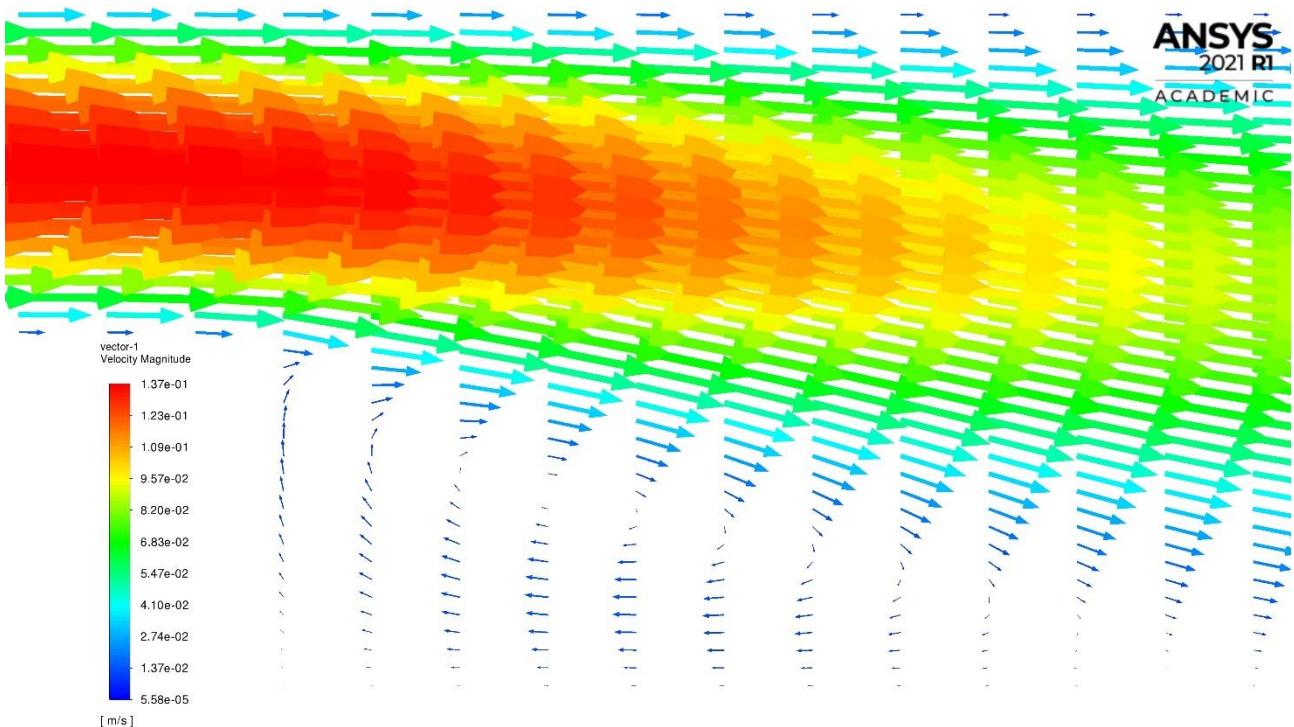


Figura 30: condizione di ricircolo: vector plot campo di velocità Fluent , $Re=100$ - mesh 120x40

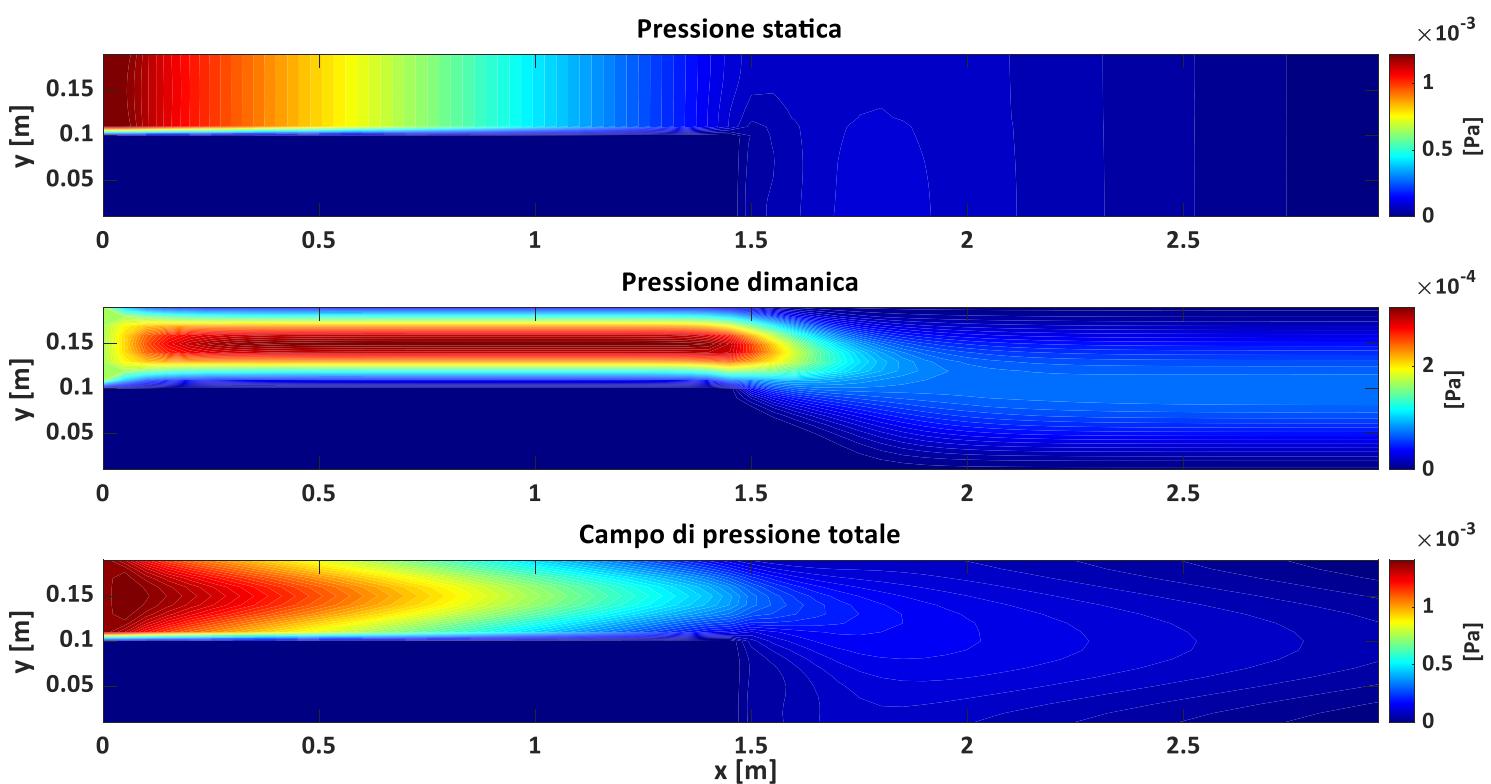


Figura 31: contour plot campi di pressione, $Re=100$ - mesh 60x20

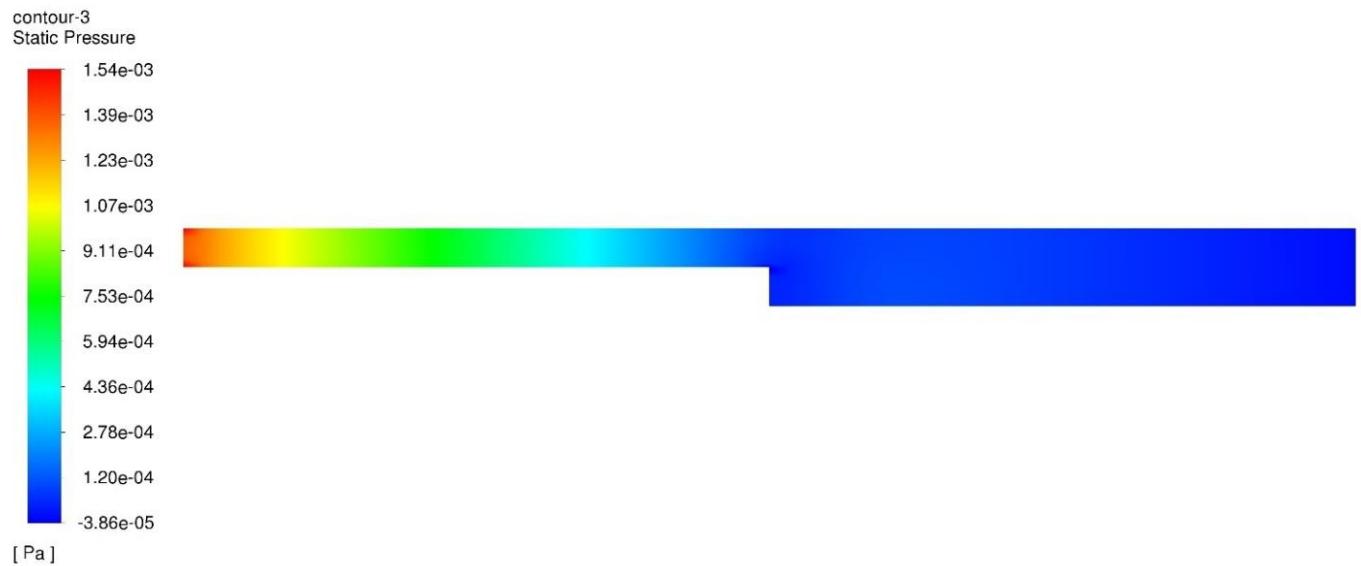


Figura 32: contour plot Fluent campo di pressione statica , $Re=100$ - mesh 60x20

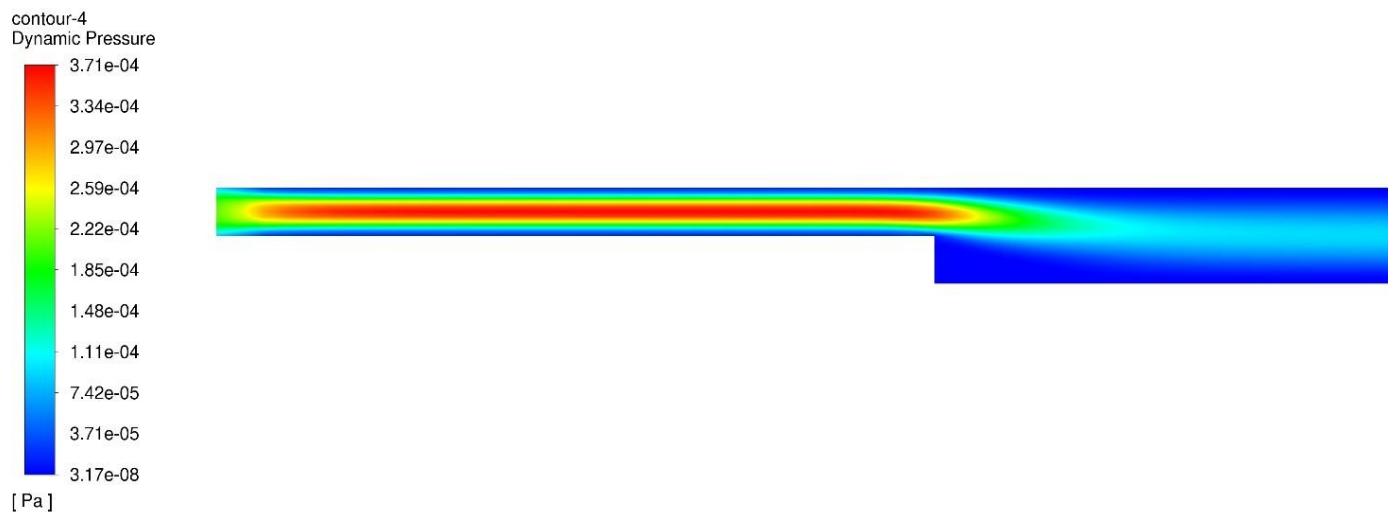


Figura 33: contour plot Fluent campo di pressione dinamica, $Re=100$ - mesh 60x20

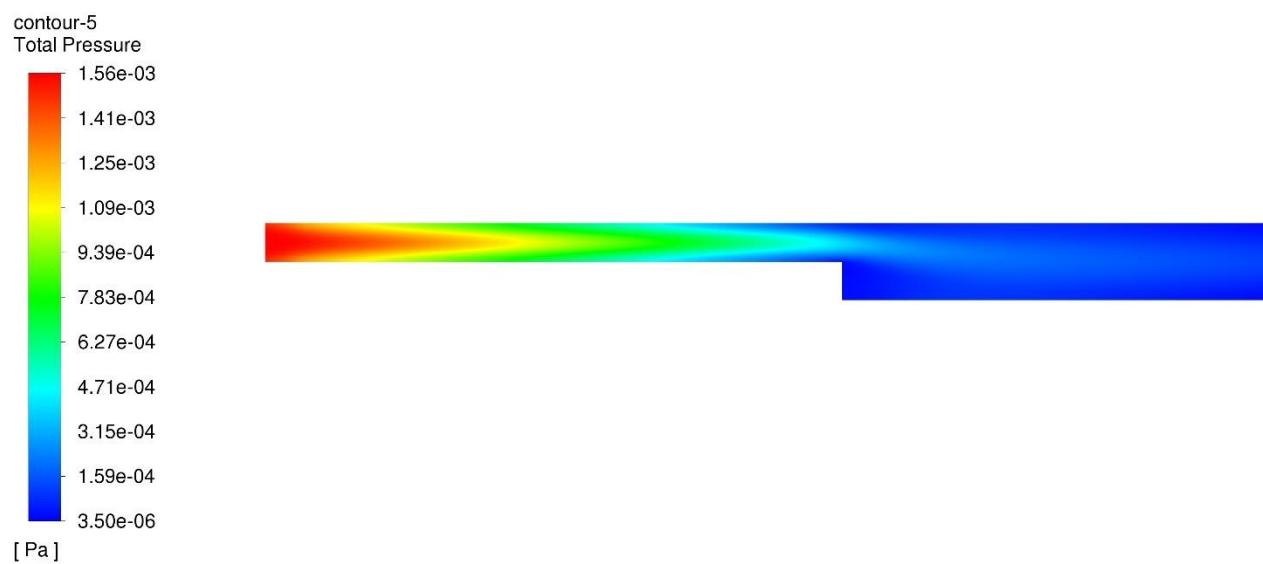


Figura 34: contour plot Fluent campo di pressione totale, $Re=100$ - mesh 60x20

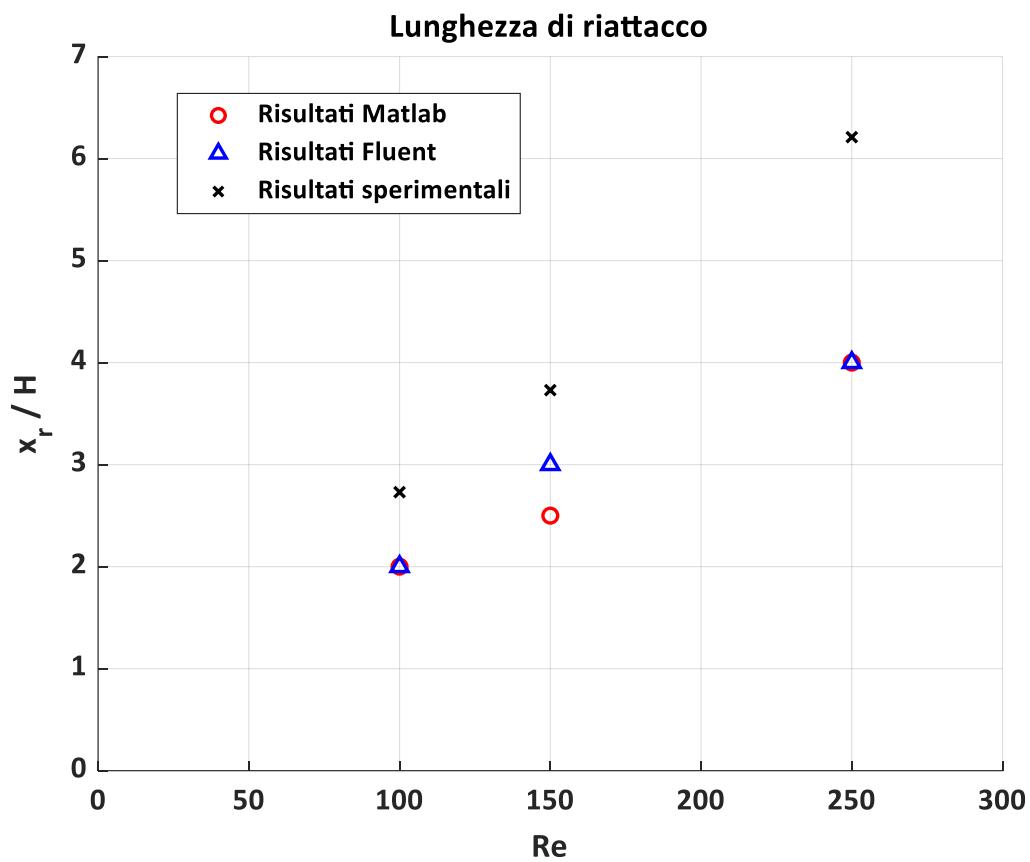


Figura 35: confronto lunghezze di riattacco al variare del Reynolds, mesh 60x20

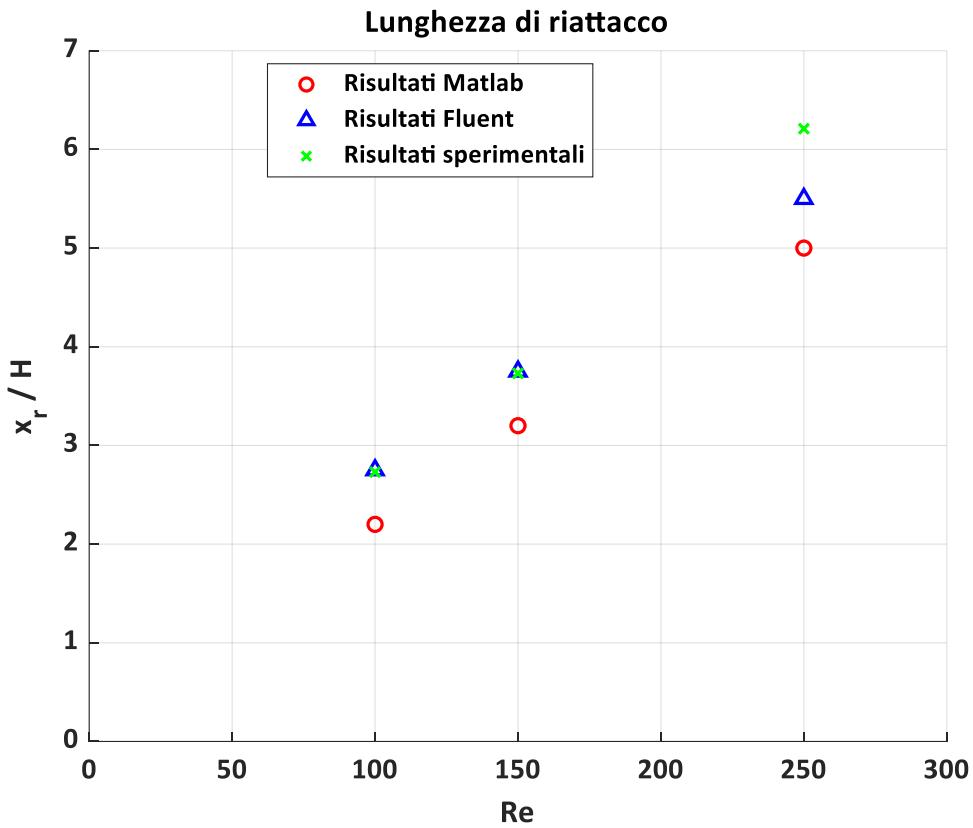


Figura 36: confronto lunghezze di riattacco al variare del Reynolds, mesh 120x40

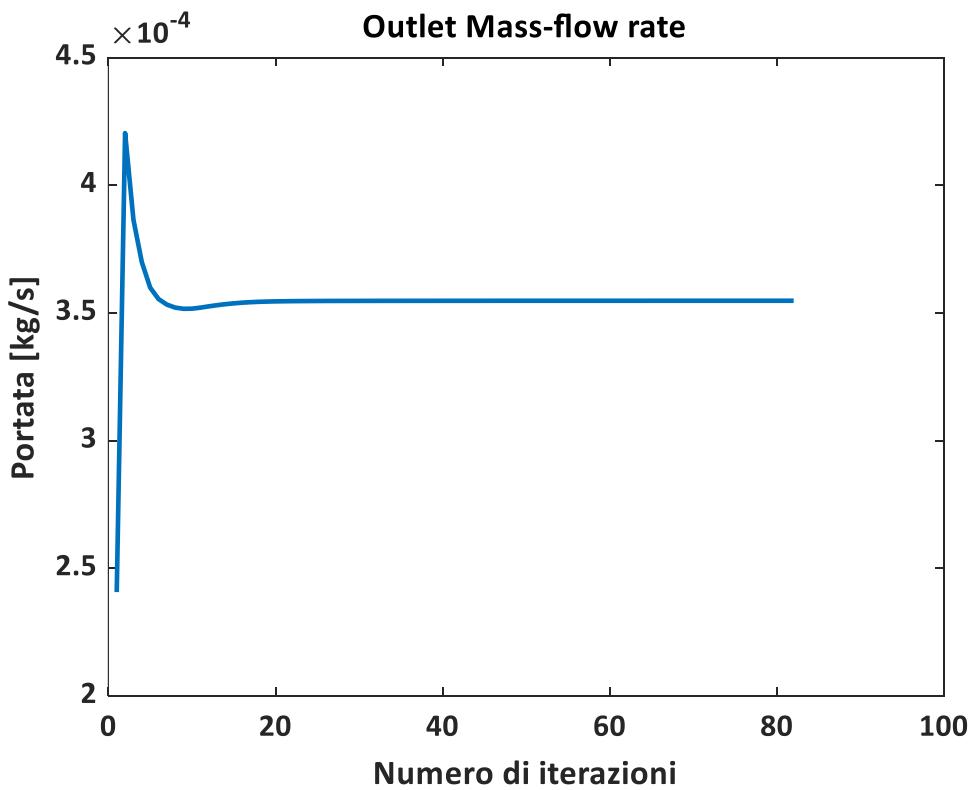


Figura 37: andamento portata outlet, Re=100 - mesh 60x20

In *tabella 11* sono riportati tutti i risultati delle simulazioni al variare dell'algoritmo, schema di interpolazione, mesh e Reynolds:

Algoritmo	Schema	Mesh	Reynolds	Iterazioni	x_r [m]	Δx_r [%]	$x_r(sper)$ [m]
SIMPLE	Upwind	60x20	100	82	0.2	-26.74	0.273
			150	98	0.25	-32.98	0.371
			250	113	0.4	-35.59	0.621
		120x40	100	177	0.22	-17.58	0.273
			150	200	0.32	-12.87	0.371
			250	304	0.5	-19.48	0.621
	Differenze centrate	60x20	100	78	0.2	-26.74	0.273
			150	111	0.3	-19.57	0.371
			250	265	0.5	-19.48	0.621
		120x40	100	177	0.25	-8.42	0.273
			150	214	0.35	-6.17	0.371
			250	228	0.55	-11.43	0.621
SIMPLER	Upwind	60x20	100	20	0.25	-8.42	0.273
			150	40	0.30	-19.57	0.371
			250	87	0.35	-43.64	0.621
		120x40	100	20	0.15	-45.05	0.273
			150	41	0.27	-26.27	0.371
			250	86	0.45	-27.54	0.621
	Differenze centrate	60x20	100	74	0.3	9.90	0.273
			150	80	0.33	-10.25	0.371
			250	147	0.55	-11.43	0.621
		120x40	100	64	0.25	-8.59	0.273
			150	93	0.34	-8.57	0.371
			250	228	0.58	-6.50	0.621

Tabella 11: riepilogo risultati caso 2

In *tavella 12* è presente il confronto delle lunghezze di riattacco ottenute in Matlab e Fluent al variare di Mesh e Reynolds:

Algoritmo	Schema	Mesh	Reynolds	Δx_r Fluent	Δx_r Matlab
SIMPLE	Upwind	60x20	100	-26.74 %	-26.74 %
			150	-20 %	-32.98 %
			250	-35.59 %	-35.59 %
		120x40	100	0.73 %	-17.58 %
			150	0.54 %	-12.87 %
			250	-11.43 %	-19.48 %

Tabella 12: confronto lunghezze di riattacco in Matlab e Fluent, SIMPLE-Upwind

5. Caso non stazionario

In questo ultimo paragrafo vengono mostrati i risultati associati al **problema non stazionario** per il *caso 2* (canale con gradino). È possibile adottare l'algoritmo SIMPLE anche nel caso di non stazionarietà del flusso, attraverso una modifica delle equazioni discretizzate della quantità di moto e l'introduzione di un ciclo esterno per tenere conto dell'avanzamento temporale del flusso stesso. In particolare, l'equazione discretizzata della quantità di moto è identica alla 2.4 ma presenta un termine aggiuntivo a_p^0 e un termine sorgente b:

$$a_p = \sum_{nb} a_{nb} + \Delta F_{ew} + \Delta F_{ns} + a_p^0 \quad (5.1)$$

$$b = u_p^0 a_p^0 \quad (5.2)$$

con:

$$u_p^0 = \frac{\Delta V}{\Delta t} = \frac{\Delta x \Delta y}{\Delta t} \quad (5.3)$$

u_p^0 è il valore nodale della griglia staggered valutato al tempo precedente (t) rispetto a quello attuale ($t + \Delta t$). Il prodotto $\Delta x \Delta y$ rappresenta il volume della cella di calcolo nel caso bidimensionale, mentre a denominatore compare l'intervallo temporale di iterazione Δt . Si deve modificare quindi l'algoritmo in *figura 5*:

1. inizializzando i campi di pressione e velocità all'istante di tempo iniziale t_{start} e scegliendo un opportuno passo di integrazione temporale Δt ;
2. calcolando i termini a_p^0 e b che si manterranno costanti fino alla successiva iterazione temporale;
3. applicando ogni passo già visto per il SIMPLE fino ad avvenuta convergenza;
4. si avanza con la soluzione di uno step temporale Δt , utilizzando come valori iniziali quelli ottenuti al termine del precedente step;
5. si procede fino al raggiungimento di $t=t_{\text{end}}$.

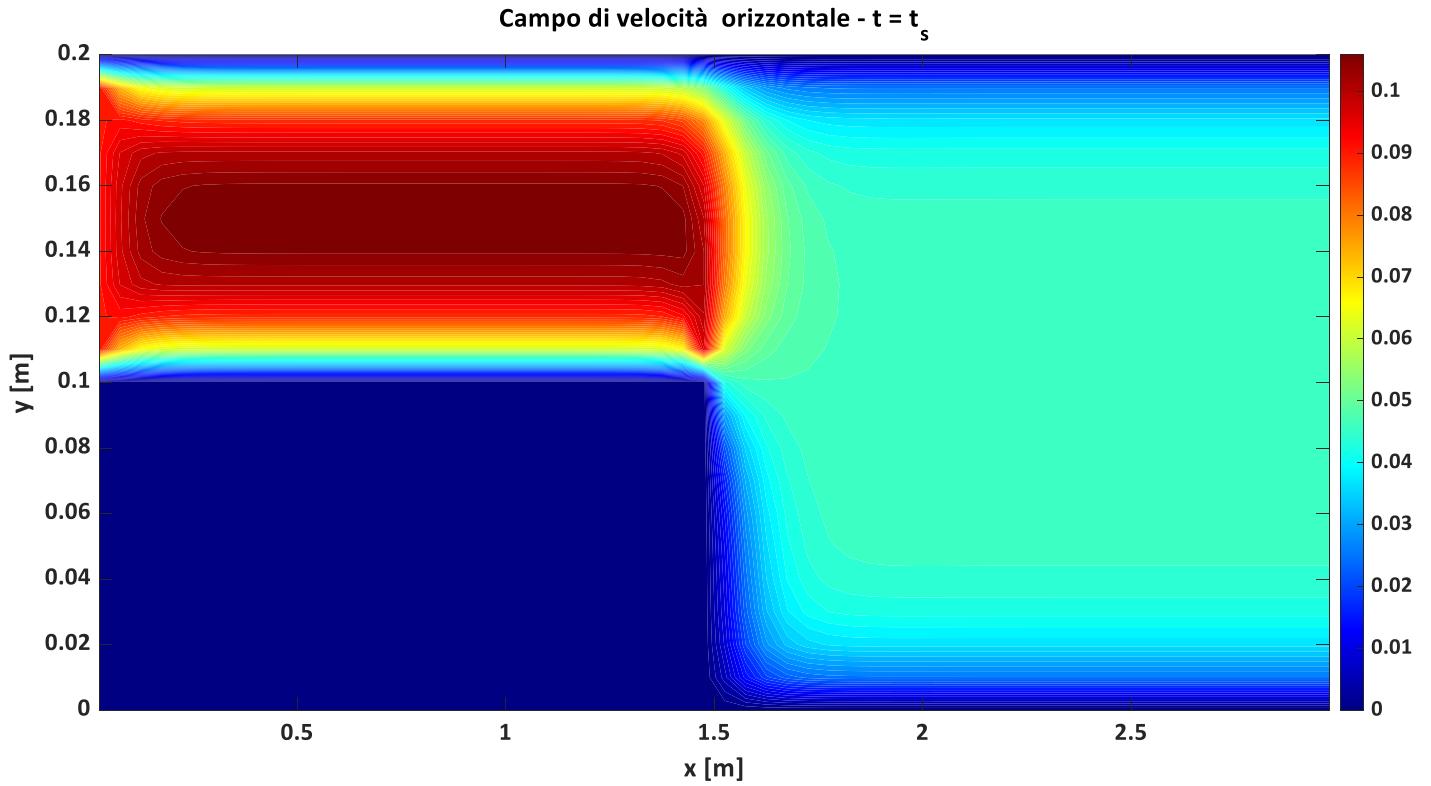


Figura 38: contour plot campo di velocità u all'inizio del ciclo iterativo - $t=t_s$, $Re=100$ – mesh 60x20 - SIMPLE

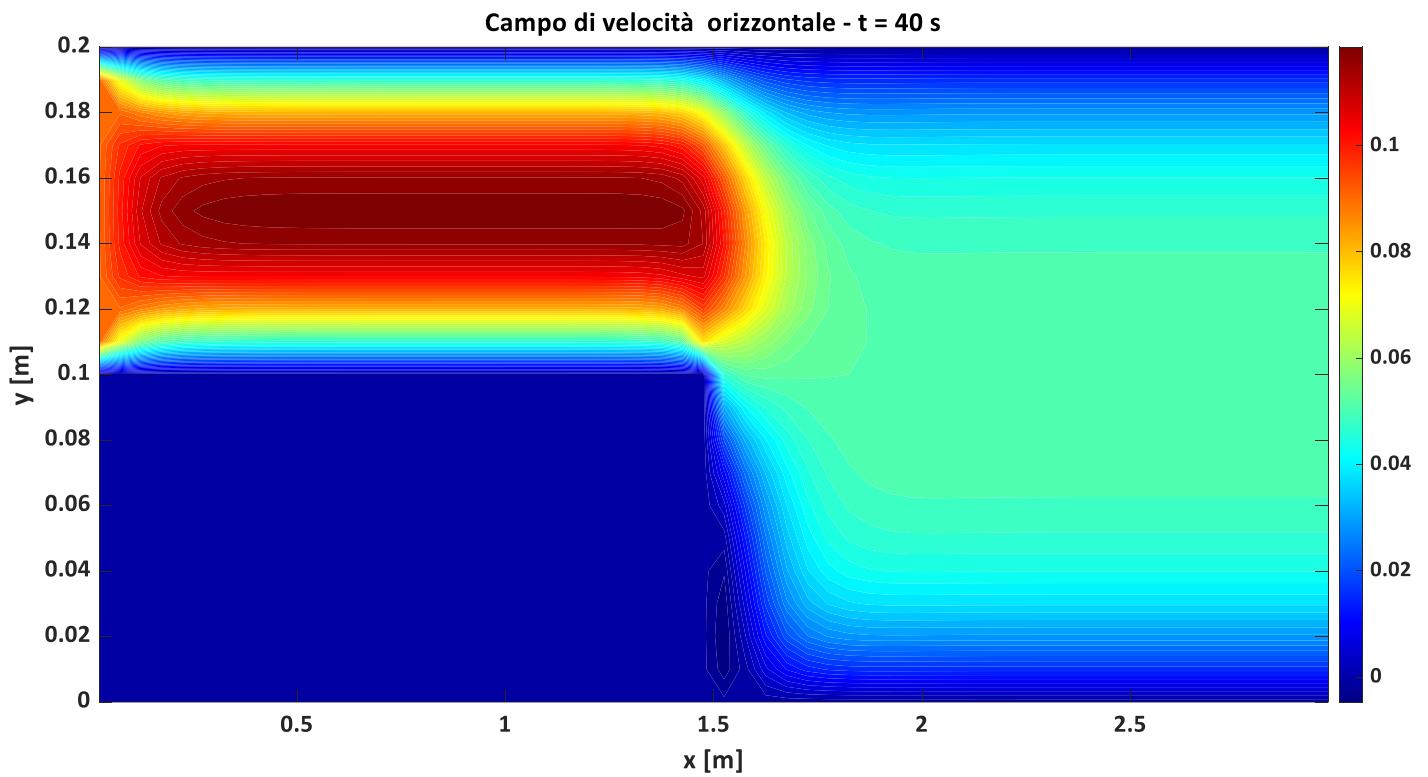


Figura 39: contour plot campo di velocità u - t=40 s , Re=100 – mesh 60x20

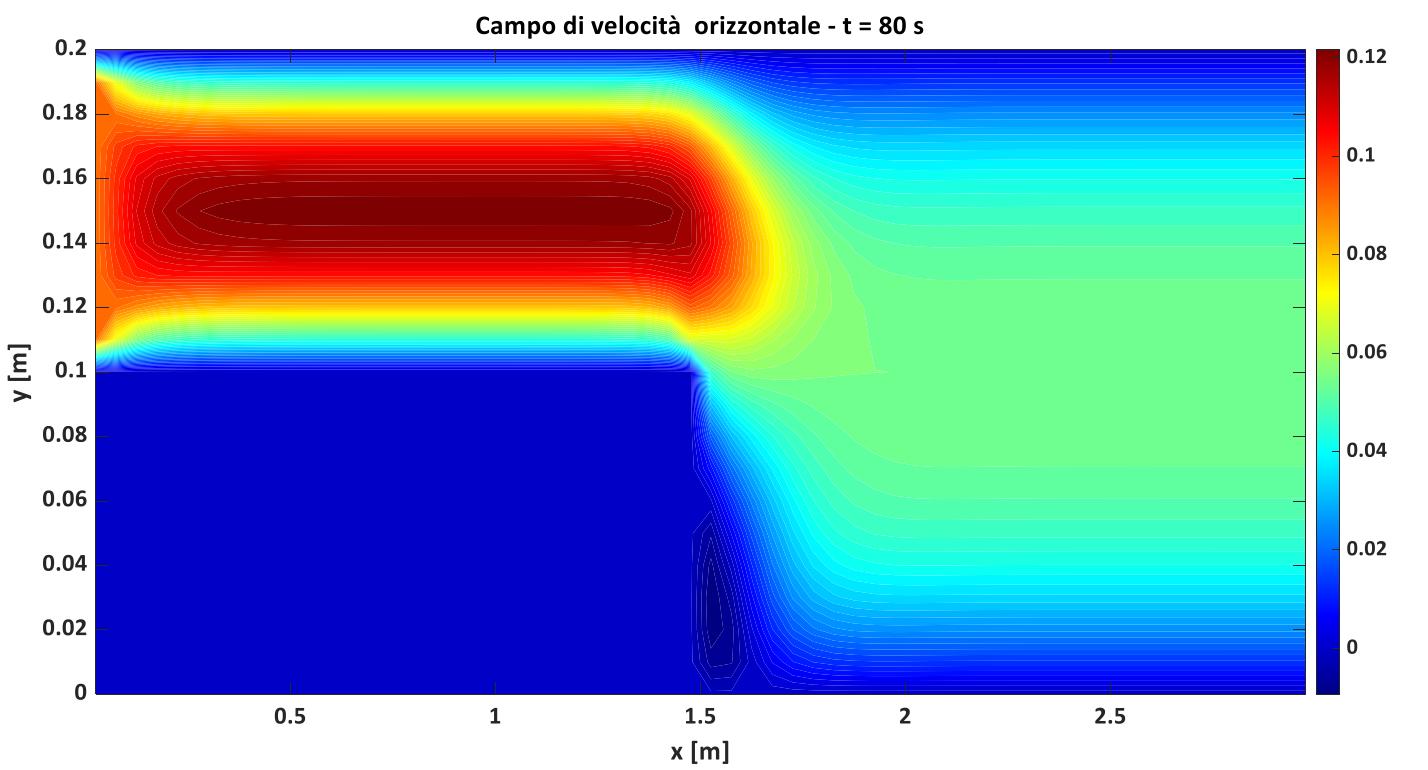


Figura 40: contour plot campo di velocità u - t=80 s , Re=100 – mesh 60x20 - SIMPLE

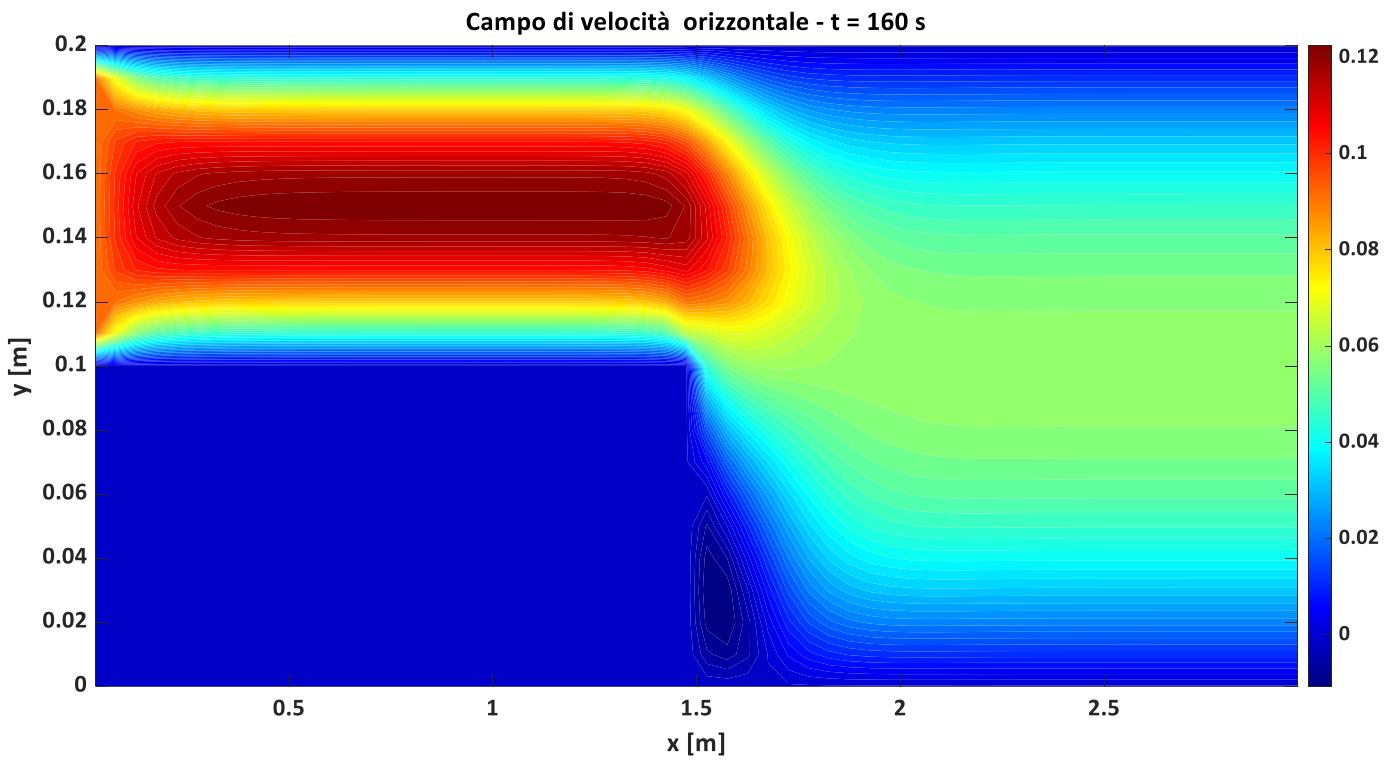


Figura 41: contour plot campo di velocità u - $t=160$ s , $Re=100$ – mesh 60x20 - SIMPLE

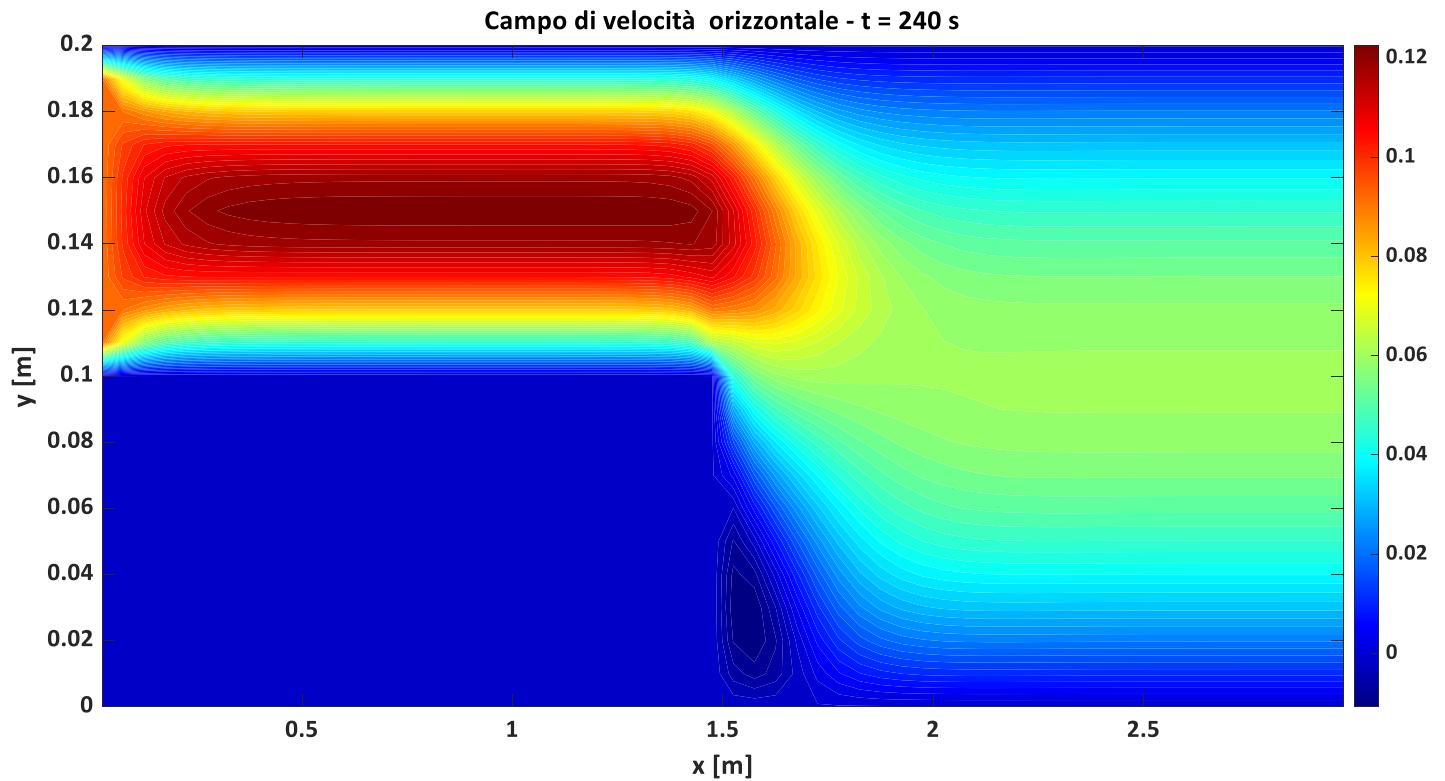


Figura 42: contour plot campo di velocità u - $t=240$ s , $Re=100$ – mesh 60x20 - SIMPLE

Le stesse modifiche sono state implementate anche all'algoritmo **SIMPLER** per adattarlo al caso non stazionario. Di seguito sono presenti i contour plot del campo di velocità verticale al variare del tempo di campionamento:

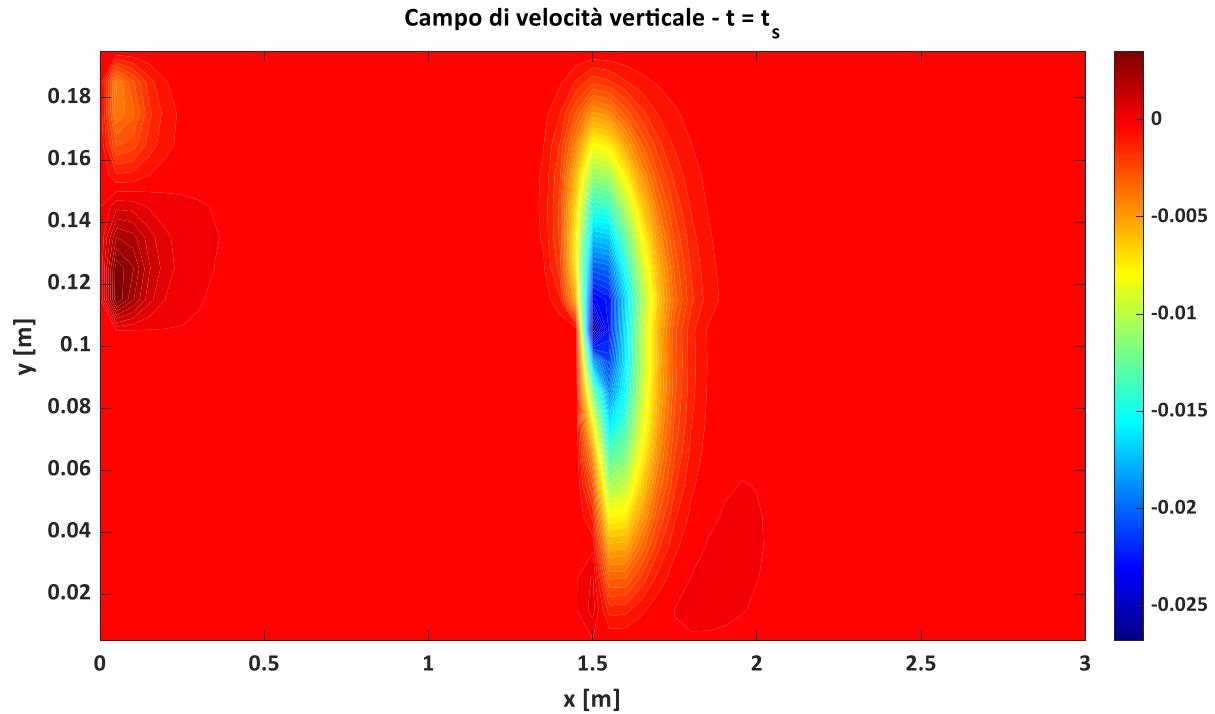


Figura 43: contour plot campo di velocità v - $t=t_s$, $Re=100$ – mesh 60×20 - SIMPLER

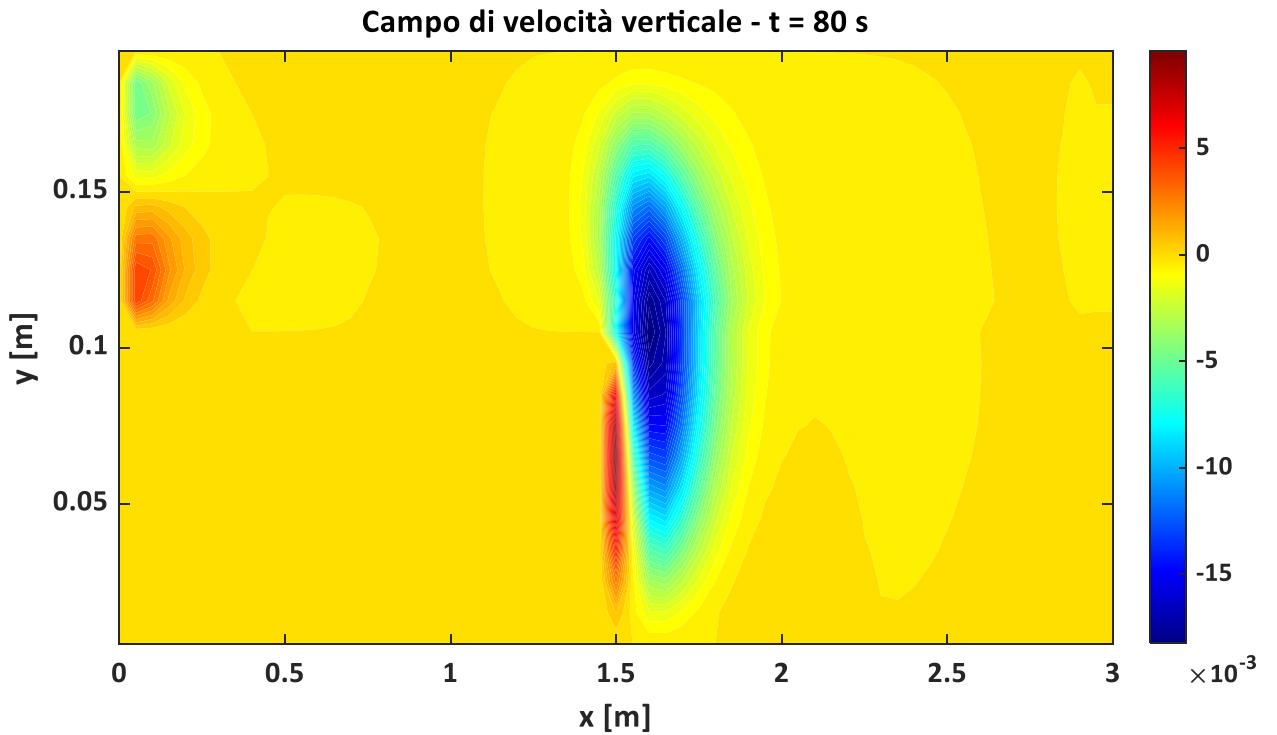


Figura 44: contour plot campo di velocità v - $t=80$ s , $Re=100$ – mesh 60×20 - SIMPLER

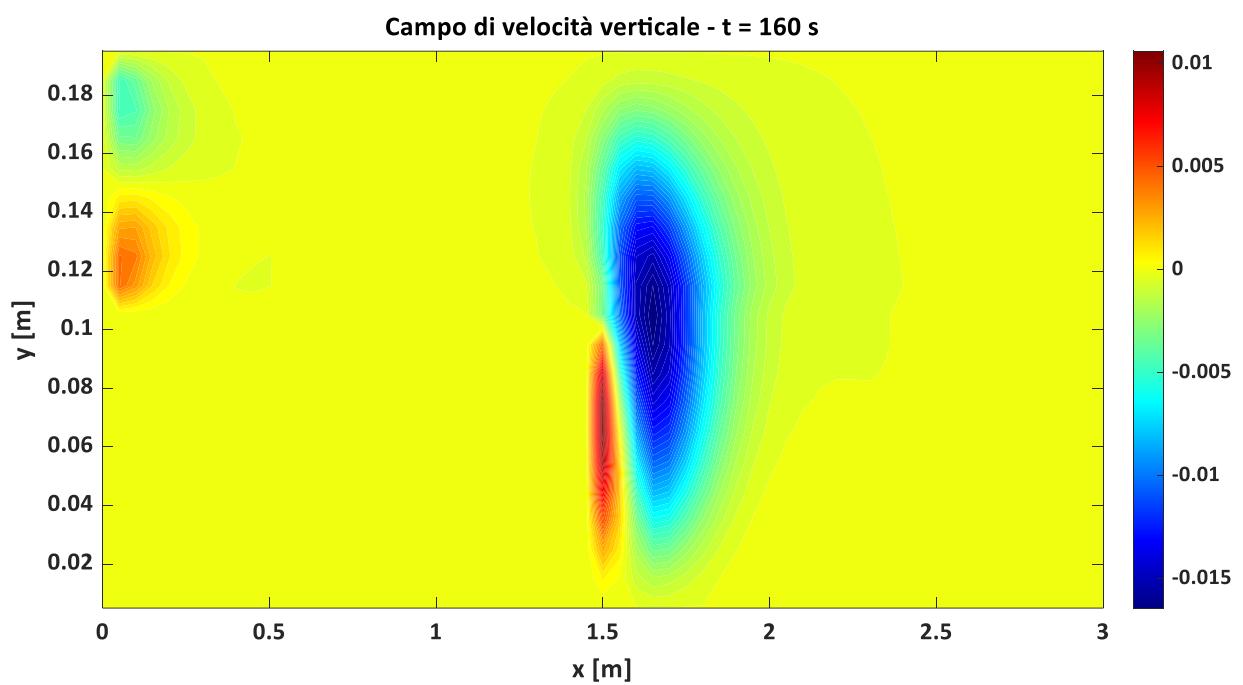


Figura 45: contour plot campo di velocità v - t=160 s , Re=100 – mesh 60x20 - SIMPLER

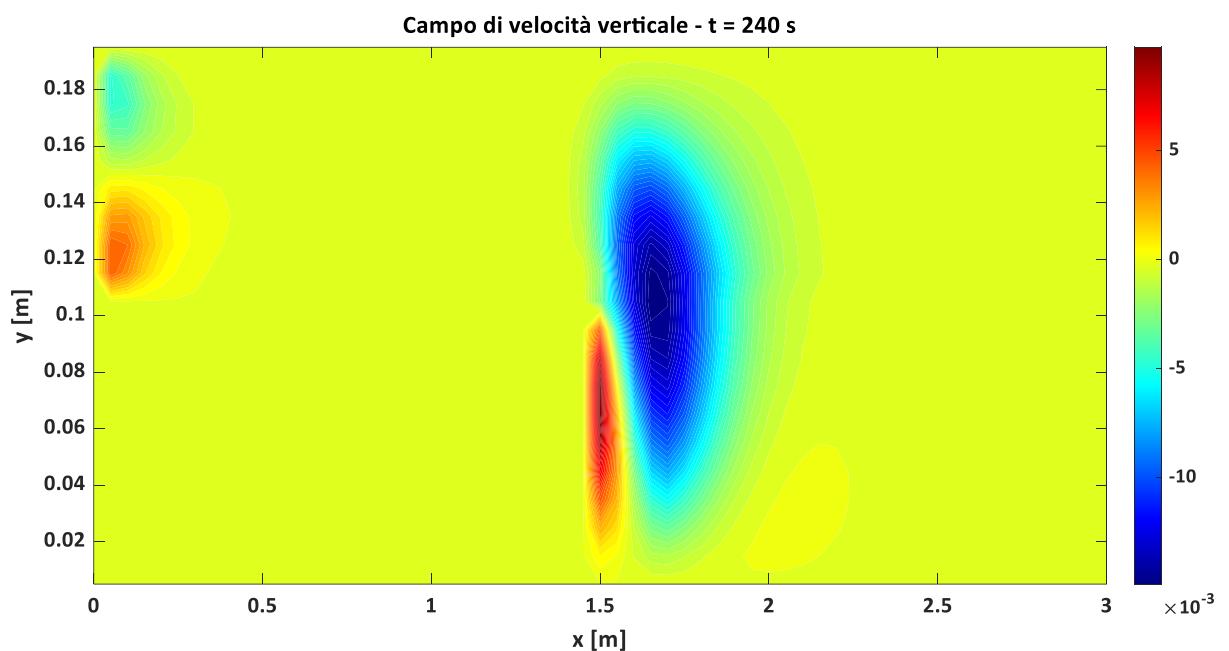


Figura 46: : contour plot campo di velocità v - t=240 s, Re=100 – mesh 60x20 - SIMPLER