

# La matematica negli Sport

Nicola Parolini



Dipartimento di Matematica  
Politecnico di Milano

**Matematica Senza Frontiere**

Monza - 3 Maggio 2019

- un interessante binomio per fare **ricerca**

- un interessante binomio per fare **ricerca**
- un campo da gioco stimolante su cui fare **innovazione**

- un interessante binomio per fare **ricerca**
- un campo da gioco stimolante su cui fare **innovazione**
- uno strumento molto efficace per fare **divulgazione**

- un interessante binomio per fare **ricerca**
- un campo da gioco stimolante su cui fare **innovazione**
- uno strumento molto efficace per fare **divulgazione**

Modelli matematici e numerici applicati a ...

- un interessante binomio per fare **ricerca**
- un campo da gioco stimolante su cui fare **innovazione**
- uno strumento molto efficace per fare **divulgazione**

Modelli matematici e numerici applicati a ...



Vela

- un interessante binomio per fare **ricerca**
- un campo da gioco stimolante su cui fare **innovazione**
- uno strumento molto efficace per fare **divulgazione**

Modelli matematici e numerici applicati a ...



Vela



Canottaggio

- un interessante binomio per fare **ricerca**
- un campo da gioco stimolante su cui fare **innovazione**
- uno strumento molto efficace per fare **divulgazione**

Modelli matematici e numerici applicati a ...



Vela



Canottaggio



Nuoto

Da dove cominciamo ...

## Da dove cominciamo ... dal golf

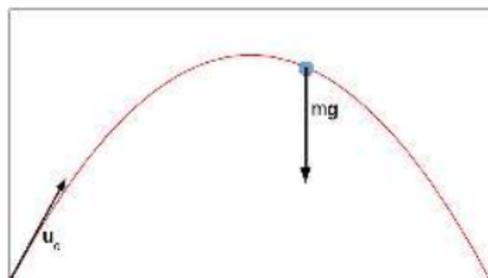


Quale **matematica** serve per calcolare la traiettoria di una palla da golf lanciata con velocità iniziale  $\mathbf{u}_0$ ?

# Da dove cominciamo ... dal golf



Quale **matematica** serve per calcolare la traiettoria di una palla da golf lanciata con velocità iniziale  $\mathbf{u}_0$ ?



**Senza forze aerodinamiche**

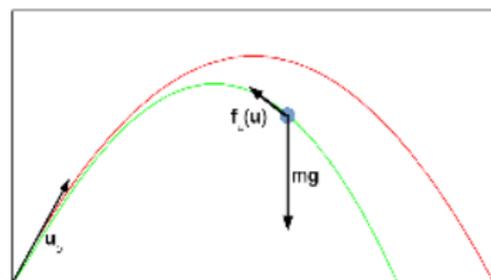
$$\mathbf{x}(t) = \mathbf{u}_0 t - \frac{1}{2} \mathbf{g} t^2$$

(relazione algebrica)

# Da dove cominciamo ... dal golf



Quale **matematica** serve per calcolare la traiettoria di una palla da golf lanciata con velocità iniziale  $\mathbf{u}_0$ ?



**Con forze aerodinamiche**

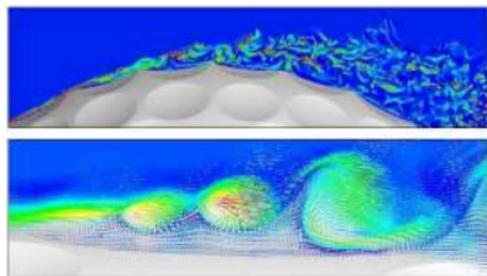
$$m\ddot{\mathbf{x}}(t) = m\mathbf{g} + \mathbf{f}_a = m\mathbf{g} + \frac{1}{2}\rho\dot{\mathbf{x}}^2 C_d S$$

(equazione differenziale ordinaria)

# Da dove cominciamo ... dal golf



Quale **matematica** serve per calcolare la traiettoria di una palla da golf lanciata con velocità iniziale  $\mathbf{u}_0$ ?

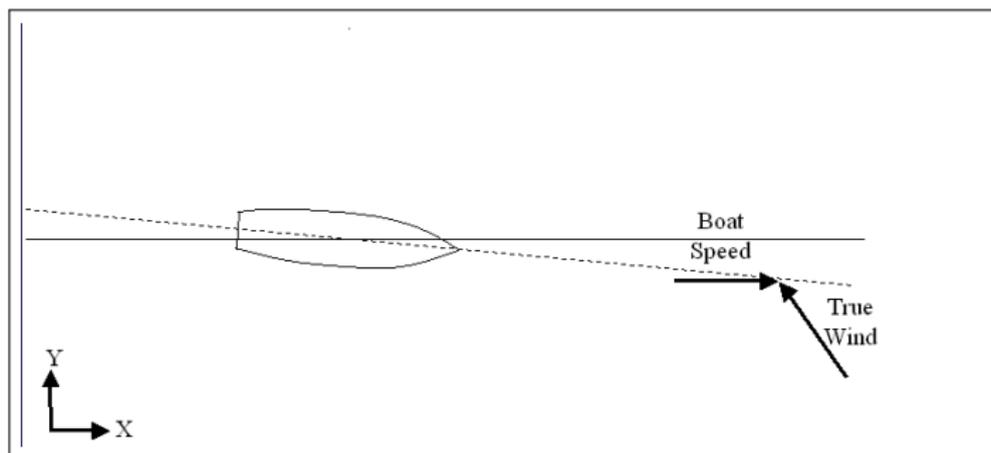


Smith et al, *Int. J. of Heat and Fluid Flow* 31, 262-273 (2012)

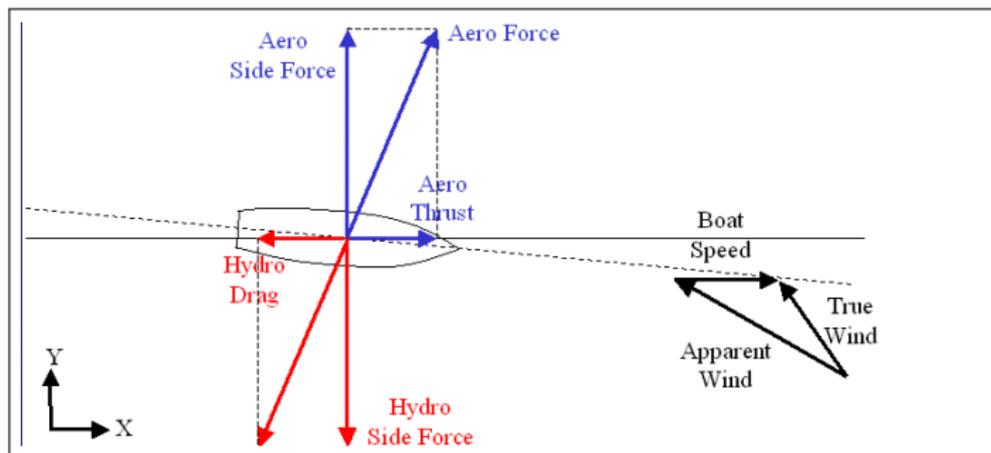
**Simulazione numerica del flusso**

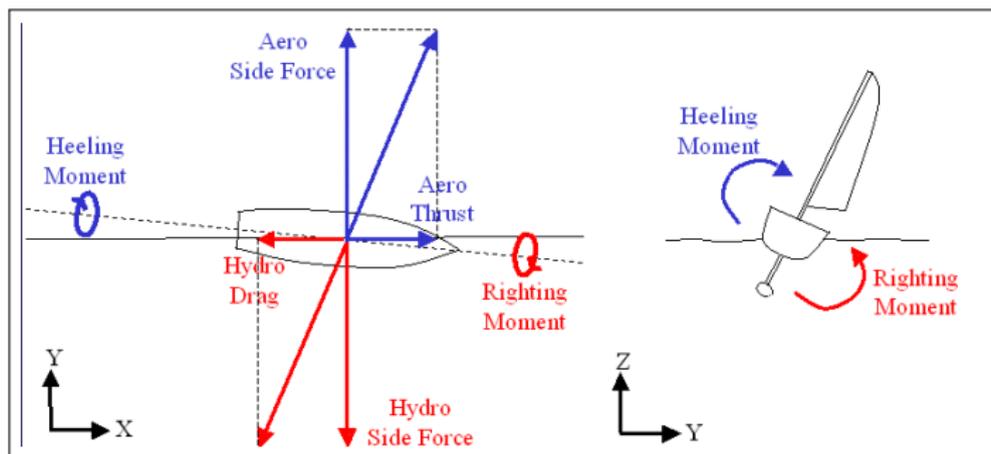
$$\partial_t \mathbf{u} + (\mathbf{u} \cdot \nabla) \mathbf{u} - \nabla \cdot \mathbf{T}(\mathbf{u}, p) = \mathbf{f},$$
$$\nabla \cdot \mathbf{u} = 0,$$

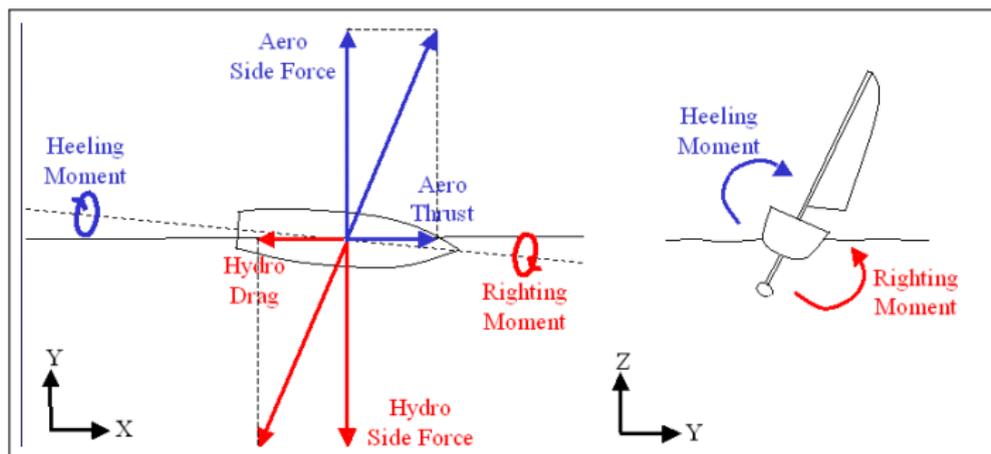
(equazioni alle derivate parziali)



# Matematica e Vela



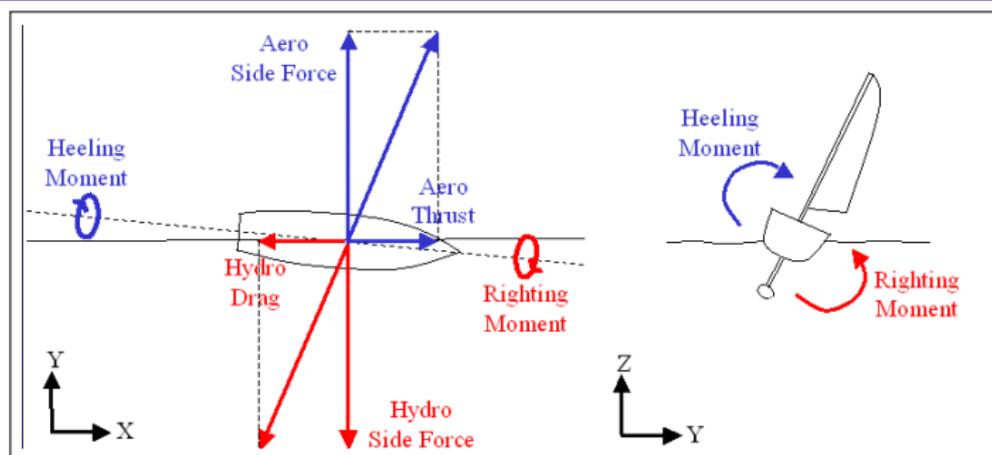




## Equilibrio

Per ogni configurazione, il VPP calcola la velocità  $V$  e l'assetto  $A$  associato allo stato di equilibrio delle forze (ODE):

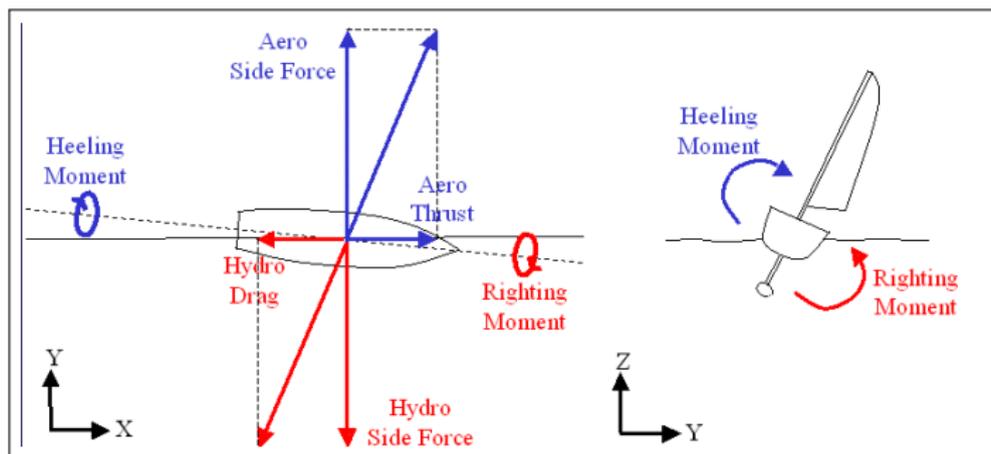
$$\begin{cases} M \ddot{X} = T_a(V, A) - D_h(V, A) \\ M \ddot{Y} = S_a(V, A) - S_h(V, A) \\ I \dot{\omega} = M_H(V, A) - M_R(V, A) \end{cases}$$



## Equilibrio

Per ogni configurazione, il VPP calcola la velocità  $V$  e l'assetto  $A$  associato allo stato di equilibrio delle forze (ODE):

$$\begin{cases} M \ddot{X} = T_a(V, A) - D_h(V, A) \\ M \ddot{Y} = S_a(V, A) - S_h(V, A) \\ I \dot{\omega} = M_H(V, A) - M_R(V, A) \end{cases} \rightarrow \begin{cases} T_a = D_h \\ S_a = S_h \\ M_H = M_R \end{cases}$$



## Equilibrio

Per ogni configurazione, il VPP calcola la velocità  $V$  e l'assetto  $A$  associato allo stato di equilibrio delle forze (ODE):

$$\begin{cases} M \ddot{X} = T_a(V, A) - D_h(V, A) \\ M \ddot{Y} = S_a(V, A) - S_h(V, A) \\ I \dot{\omega} = M_H(V, A) - M_R(V, A) \end{cases} \longrightarrow \begin{cases} T_a = D_h \\ S_a = S_h \\ M_H = M_R \end{cases} \longrightarrow V_{Eq}, A_{Eq}$$

# Come stimare le forze in gioco

# Come stimare le forze in gioco

## Vasca navale

Resistenza sullo scafo



# Come stimare le forze in gioco

## Vasca navale

Resistenza sullo scafo

## Galleria del vento

Forze su vele e appendici



# Come stimare le forze in gioco

## Vasca navale

Resistenza sullo scafo

## Galleria del vento

Forze su vele e appendici

## Test in mare

Misure a scala reale



10/10/11 © 2011 Mercedes-Benz F1 Team. All rights reserved. 10/10/11

# Come stimare le forze in gioco

## Vasca navale

Resistenza sullo scafo

## Galleria del vento

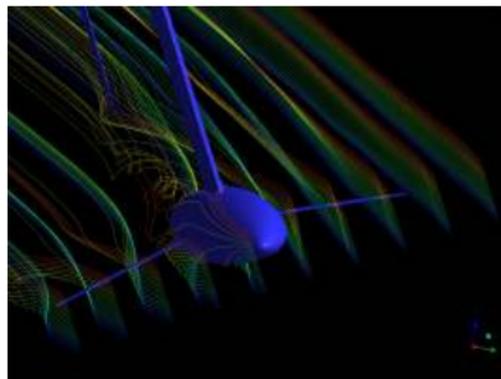
Forze su vele e appendici

## Simulazioni CFD

Forze e visualizzazioni

## Test in mare

Misure a scala reale



# Come stimare le forze in gioco

## Vasca navale

Resistenza sullo scafo

## Galleria del vento

Forze su vele e appendici

## Simulazioni CFD

Forze e visualizzazioni

## Test in mare

Misure a scala reale

## Per ogni possibile configurazione di progetto:

- Simulazioni e misure sperimentali in un numero limitato di velocità e assetti  $F(V_i, A_j)$ ;
- Regressione dei dati sul range di parametri  $F(V, A)$ ,  
( $V_{\min} < V < V_{\max}$ ,  $A_{\min} < A < A_{\max}$ );
- Calcolo delle prestazioni  $V_{Eq}$ ,  $A_{Eq}$  con il Velocity Prediction Program (VPP).

# Come stimare le forze in gioco

Vasca navale

Resistenza sullo scafo

Galleria del vento

Forze su vele e appendici

VPP

Simulazioni CFD

Forze e visualizzazioni

Test in mare

Misure a scala reale

Per ogni possibile configurazione di progetto:

- Simulazioni e misure sperimentali in un numero limitato di velocità e assetti  $F(V_i, A_j)$ ;
- Regressione dei dati sul range di parametri  $F(V, A)$ ,  $(V_{\min} < V < V_{\max}, A_{\min} < A < A_{\max})$ ;
- Calcolo delle prestazioni  $V_{Eq}, A_{Eq}$  con il Velocity Prediction Program (VPP).

# Un problema matematico complesso da discretizzare

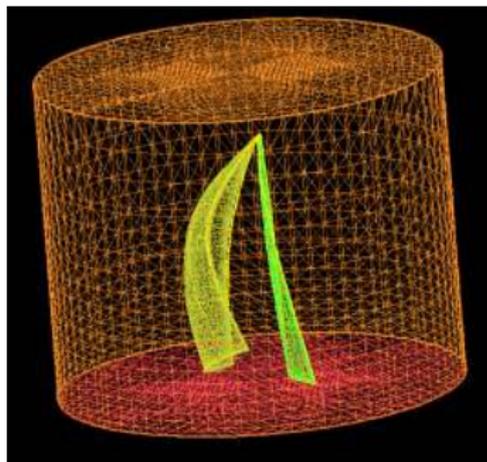
- Le equazioni di Navier-Stokes non si possono risolvere analiticamente

# Un problema matematico complesso da discretizzare

- Le equazioni di Navier-Stokes non si possono risolvere analiticamente
- Discretizzazione del problema (dal continuo al discreto)

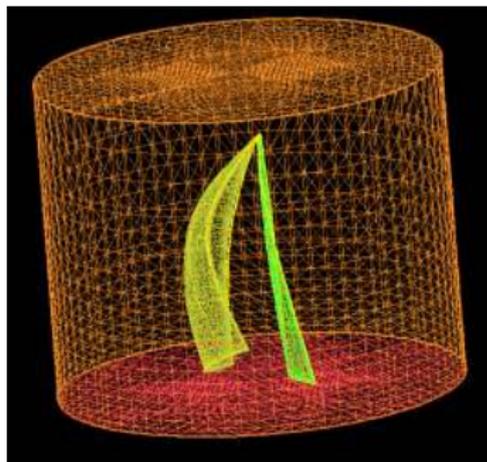
# Un problema matematico complesso da discretizzare

- Le equazioni di Navier-Stokes non si possono risolvere analiticamente
- Discretizzazione del problema (dal continuo al discreto)
- Decomposizione del dominio fisico in una **griglia computazionale**



# Un problema matematico complesso da discretizzare

- Le equazioni di Navier-Stokes non si possono risolvere analiticamente
- Discretizzazione del problema (dal continuo al discreto)
- Decomposizione del dominio fisico in una **griglia computazionale**
- La soluzione del problema discretizzato è il valore dell'incognite (velocità, pressione, ...) in un numero finito (ma **molto grande**) di punti e di istanti temporali



$$\mathbf{u}(\mathbf{x}, t) \longrightarrow \mathbf{u}(\mathbf{x}_i, t^n)$$

$$\frac{\partial \mathbf{u}}{\partial t}(\mathbf{x}, t) \longrightarrow \frac{\mathbf{u}(\mathbf{x}_i, t^{n+1}) - \mathbf{u}(\mathbf{x}_i, t^n)}{dt}$$

$$\frac{\partial p}{\partial x}(\mathbf{x}, t) \longrightarrow \frac{p(\mathbf{x}_i + d\mathbf{x}, t^n) - p(\mathbf{x}_i - d\mathbf{x}, t^n)}{2d\mathbf{x}}$$

# Problema discreto

- Il problema è sempre ricondotto a un **sistema lineare**

- Il problema è sempre ricondotto a un **sistema lineare**

**Esempio di sistema lineare:** trovare  $x_1$  e  $x_2$  tali che

$$\begin{cases} -x_1 + 2x_2 = 0 \\ 2x_1 - 3x_2 = 1 \end{cases} \quad 2 \text{ equazioni e } 2 \text{ incognite}$$

- Il problema è sempre ricondotto a un **sistema lineare**

**Esempio di sistema lineare:** trovare  $x_1$  e  $x_2$  tali che

$$\begin{cases} -x_1 + 2x_2 = 0 \\ 2x_1 - 3x_2 = 1 \end{cases} \quad 2 \text{ equazioni e } 2 \text{ incognite}$$

**Soluzione:**

- Il problema è sempre ricondotto a un **sistema lineare**

**Esempio di sistema lineare:** trovare  $x_1$  e  $x_2$  tali che

$$\begin{cases} -x_1 + 2x_2 = 0 \\ 2x_1 - 3x_2 = 1 \end{cases} \quad 2 \text{ equazioni e } 2 \text{ incognite}$$

**Soluzione:** isolo  $x_1$  nella prima equazione

$$x_1 = 2x_2$$

- Il problema è sempre ricondotto a un **sistema lineare**

**Esempio di sistema lineare:** trovare  $x_1$  e  $x_2$  tali che

$$\begin{cases} -x_1 + 2x_2 = 0 \\ 2x_1 - 3x_2 = 1 \end{cases} \quad \text{2 equazioni e 2 incognite}$$

**Soluzione:** isolo  $x_1$  nella prima equazione e sostituisco nella seconda

$$x_1 = 2x_2$$

$$2(2x_2) - 3x_2 = 1$$

- Il problema è sempre ricondotto a un **sistema lineare**

**Esempio di sistema lineare:** trovare  $x_1$  e  $x_2$  tali che

$$\begin{cases} -x_1 + 2x_2 = 0 \\ 2x_1 - 3x_2 = 1 \end{cases} \quad \text{2 equazioni e 2 incognite}$$

**Soluzione:** isolo  $x_1$  nella prima equazione e sostituisco nella seconda

$$x_1 = 2x_2$$

$$2(2x_2) - 3x_2 = 1 \quad \longrightarrow \quad \mathbf{x_2 = 1}$$

- Nel nostro caso abbiamo milioni di equazioni e **milioni di incognite**

# Problema discreto

- Il problema è sempre ricondotto a un **sistema lineare**

**Esempio di sistema lineare:** trovare  $x_1$  e  $x_2$  tali che

$$\begin{cases} -x_1 + 2x_2 = 0 \\ 2x_1 - 3x_2 = 1 \end{cases} \quad \text{2 equazioni e 2 incognite}$$

**Soluzione:** isolo  $x_1$  nella prima equazione e sostituisco nella seconda

$$x_1 = 2x_2 \quad \longrightarrow \quad \mathbf{x_1 = 2}$$

$$2(2x_2) - 3x_2 = 1 \quad \longrightarrow \quad \mathbf{x_2 = 1}$$

## E se le incognite sono di più ?

La *regola di Cramer* fornisce esplicitamente la soluzione per qualunque numero di incognite.

## E se le incognite sono di più ?

La *regola di Cramer* fornisce esplicitamente la soluzione per qualunque numero di incognite.

Quanto costa ?  $\mathcal{O}((N + 1)!) \text{ operazioni}$

## E se le incognite sono di più ?

La *regola di Cramer* fornisce esplicitamente la soluzione per qualunque numero di incognite.

Quanto costa ?  $\mathcal{O}((N + 1)!) \text{ operazioni}$

Dimensioni del sistema	→	Numero di operazioni
$2 \times 2$		6

## E se le incognite sono di più ?

La *regola di Cramer* fornisce esplicitamente la soluzione per qualunque numero di incognite.

Quanto costa ?  $\mathcal{O}((N + 1)!) \text{ operazioni}$

Dimensioni del sistema		Numero di operazioni
$2 \times 2$	→	6
$3 \times 3$	→	24

# E se le incognite sono di più ?

La *regola di Cramer* fornisce esplicitamente la soluzione per qualunque numero di incognite.

Quanto costa ?  $\mathcal{O}((N + 1)!) \text{ operazioni}$

Dimensioni del sistema		Numero di operazioni
$2 \times 2$	→	6
$3 \times 3$	→	24
$5 \times 5$	→	720

## E se le incognite sono di più ?

La *regola di Cramer* fornisce esplicitamente la soluzione per qualunque numero di incognite.

Quanto costa ?  $\mathcal{O}((N + 1)!) \text{ operazioni}$

Dimensioni del sistema		Numero di operazioni
$2 \times 2$	→	6
$3 \times 3$	→	24
$5 \times 5$	→	720
$10 \times 10$	→	$4 \times 10^7$

# E se le incognite sono di più ?

La *regola di Cramer* fornisce esplicitamente la soluzione per qualunque numero di incognite.

Quanto costa ?  $\mathcal{O}((N + 1)!) \text{ operazioni}$

Dimensioni del sistema		Numero di operazioni
$2 \times 2$	→	6
$3 \times 3$	→	24
$5 \times 5$	→	720
$10 \times 10$	→	$4 \times 10^7$
$25 \times 25$	→	$1 \times 10^{27}$

## E se le incognite sono di più ?

La *regola di Cramer* fornisce esplicitamente la soluzione per qualunque numero di incognite.

Quanto costa ?  $\mathcal{O}((N + 1)!) \text{ operazioni}$

Dimensioni del sistema		Numero di operazioni
$2 \times 2$	→	6
$3 \times 3$	→	24
$5 \times 5$	→	720
$10 \times 10$	→	$4 \times 10^7$
$25 \times 25$	→	$1 \times 10^{27}$
$100 \times 100$	→	$9 \times 10^{159}$

# E se le incognite sono di più ?

La *regola di Cramer* fornisce esplicitamente la soluzione per qualunque numero di incognite.

Quanto costa ?  $\mathcal{O}((N + 1)!)$  operazioni

Dimensioni del sistema		Numero di operazioni
$2 \times 2$	→	6
$3 \times 3$	→	24
$5 \times 5$	→	720
$10 \times 10$	→	$4 \times 10^7$
$25 \times 25$	→	$1 \times 10^{27}$
$100 \times 100$	→	$9 \times 10^{159}$



- Tianhe2 (National Super Computer Center in Guangzhou, Cina) è il computer più veloce del mondo

# E se le incognite sono di più ?

La *regola di Cramer* fornisce esplicitamente la soluzione per qualunque numero di incognite.

Quanto costa ?  $\mathcal{O}((N + 1)!)$  operazioni

Dimensioni del sistema		Numero di operazioni
$2 \times 2$	→	6
$3 \times 3$	→	24
$5 \times 5$	→	720
$10 \times 10$	→	$4 \times 10^7$
$25 \times 25$	→	$1 \times 10^{27}$
$100 \times 100$	→	$9 \times 10^{159}$



- Tianhe2 (National Super Computer Center in Guangzhou, Cina) è il computer più veloce del mondo
- Esegue 33 862 TeraFLOPS (ovvero circa  $34 \times 10^{15}$  operazioni al secondo)

# E se le incognite sono di più ?

La *regola di Cramer* fornisce esplicitamente la soluzione per qualunque numero di incognite.

Quanto costa ?  $\mathcal{O}((N + 1)!)$  operazioni

Dimensioni del sistema		Numero di operazioni
$2 \times 2$	→	6
$3 \times 3$	→	24
$5 \times 5$	→	720
$10 \times 10$	→	$4 \times 10^7$
$25 \times 25$	→	$1 \times 10^{27}$
$100 \times 100$	→	$9 \times 10^{159}$



- Tianhe2 (National Super Computer Center in Guangzhou, Cina) è il computer più veloce del mondo
- Esegue 33 862 TeraFLOPS (ovvero circa  $34 \times 10^{15}$  operazioni al secondo)
- Ci metterebbe **1000 anni** a risolvere un sistema  $25 \times 25$  con la regola di Cramer!

## Ma i sistemi veri sono piu' grandi ...

- Una simulazione tipica per lo scafo arriva ad avere sistemi di dimensione  $N = 200\,000\,000$
- Con il *metodo di eliminazione di Gauss* si ottiene una soluzione in  $\mathcal{O}(N^3)$  operazioni

# Ma i sistemi veri sono piu' grandi ...

- Una simulazione tipica per lo scafo arriva ad avere sistemi di dimensione  $N = 200\,000\,000$
- Con il *metodo di eliminazione di Gauss* si ottiene una soluzione in  $\mathcal{O}(N^3)$  operazioni

Dimensioni del sistema

Numero di operazioni

# Ma i sistemi veri sono piu' grandi ...

- Una simulazione tipica per lo scafo arriva ad avere sistemi di dimensione  $N = 200\,000\,000$
- Con il *metodo di eliminazione di Gauss* si ottiene una soluzione in  $\mathcal{O}(N^3)$  operazioni

Dimensioni del sistema  
 $10 \times 10$



Numero di operazioni  
 $4 \times 10^2$

## Ma i sistemi veri sono piu' grandi ...

- Una simulazione tipica per lo scafo arriva ad avere sistemi di dimensione  $N = 200\,000\,000$
- Con il *metodo di eliminazione di Gauss* si ottiene una soluzione in  $\mathcal{O}(N^3)$  operazioni

Dimensioni del sistema		Numero di operazioni
$10 \times 10$	→	$4 \times 10^2$
$100 \times 100$	→	$3 \times 10^5$

## Ma i sistemi veri sono piu' grandi ...

- Una simulazione tipica per lo scafo arriva ad avere sistemi di dimensione  $N = 200\,000\,000$
- Con il *metodo di eliminazione di Gauss* si ottiene una soluzione in  $\mathcal{O}(N^3)$  operazioni

Dimensioni del sistema		Numero di operazioni
$10 \times 10$	→	$4 \times 10^2$
$100 \times 100$	→	$3 \times 10^5$
$10\,000 \times 10\,000$	→	$3 \times 10^{11}$

## Ma i sistemi veri sono piu' grandi ...

- Una simulazione tipica per lo scafo arriva ad avere sistemi di dimensione  $N = 200\,000\,000$
- Con il *metodo di eliminazione di Gauss* si ottiene una soluzione in  $\mathcal{O}(N^3)$  operazioni

Dimensioni del sistema		Numero di operazioni
$10 \times 10$	→	$4 \times 10^2$
$100 \times 100$	→	$3 \times 10^5$
$10\,000 \times 10\,000$	→	$3 \times 10^{11}$
$200\,000\,000 \times 200\,000\,000$	→	$3 \times 10^{24}$

# Ma i sistemi veri sono piu' grandi ...

- Una simulazione tipica per lo scafo arriva ad avere sistemi di dimensione  $N = 200\,000\,000$
- Con il *metodo di eliminazione di Gauss* si ottiene una soluzione in  $\mathcal{O}(N^3)$  operazioni

Dimensioni del sistema		Numero di operazioni
$10 \times 10$	→	$4 \times 10^2$
$100 \times 100$	→	$3 \times 10^5$
$10\,000 \times 10\,000$	→	$3 \times 10^{11}$
$200\,000\,000 \times 200\,000\,000$	→	$3 \times 10^{24}$



- Tianhe2 ci metterebbe comunque **3 anni** a risolvere il sistema necessario per una simulazione dello scafo

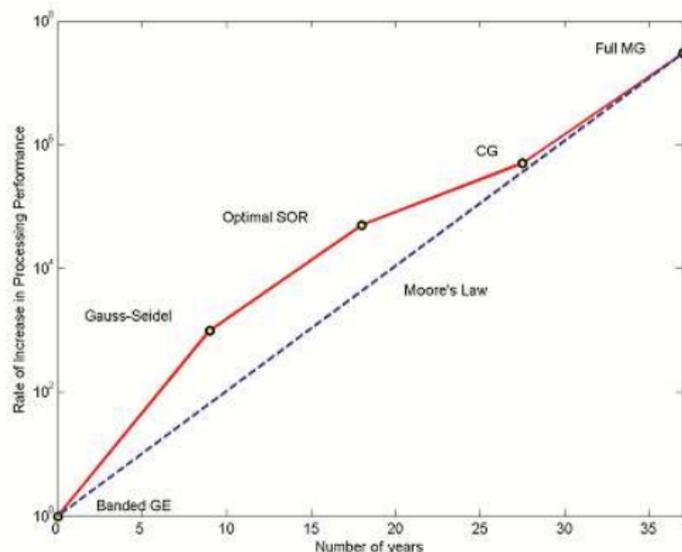
- La velocità di elaborazione dei calcolatori è cresciuta in modo esponenziale

# Hardware + Software

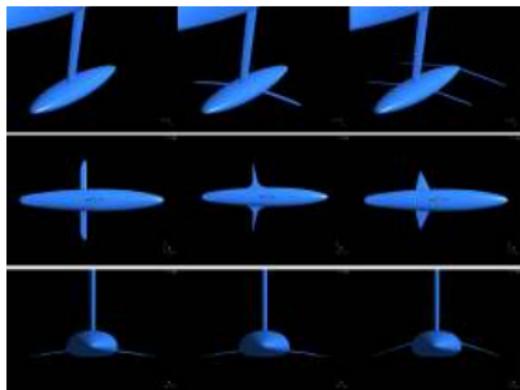
- La velocità di elaborazione dei calcolatori è cresciuta in modo esponenziale
- Parallelamente, si sono sviluppati algoritmi sempre più efficaci

# Hardware + Software

- La velocità di elaborazione dei calcolatori è cresciuta in modo esponenziale
- Parallelamente, si sono sviluppati algoritmi sempre più efficaci

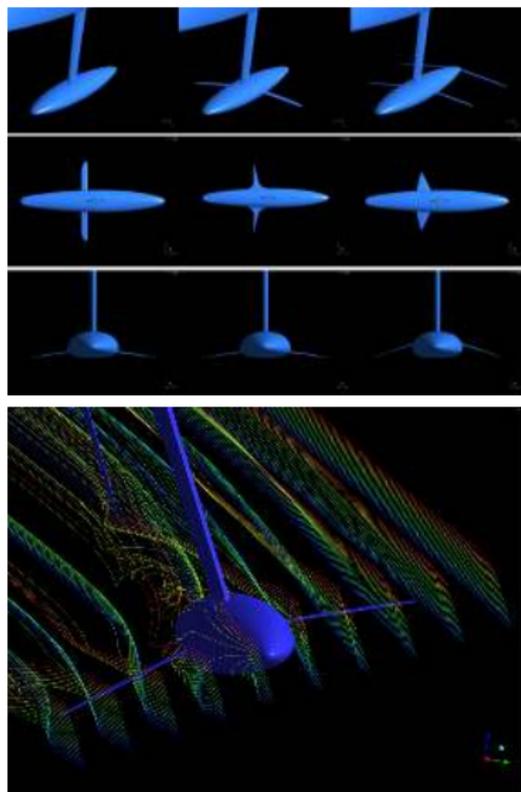


# Progetto delle appendici: bulbo, chiglia e alette



- Simulazione del flusso attorno alle appendici in diverse condizioni di regata
- Studi parametrici per diverse scelte di progetto
- Analisi di soluzioni radicalmente innovative

# Progetto delle appendici: bulbo, chiglia e alette



- Simulazione del flusso attorno alle appendici in diverse condizioni di regata
  - Studi parametrici per diverse scelte di progetto
  - Analisi di soluzioni radicalmente innovative
- 
- Necessarie griglie di calcolo molto raffinate
  - Utilizzo di visualizzazioni complesse per individuare strutture fluidodinamiche di interesse

# Dinamica (rigida) dell'imbarcazione

Seconda legge di Newton (**forza = massa x accelerazione**)

$$m\ddot{\mathbf{X}}_G = \mathbf{F} \quad (\text{lineare})$$

# Dinamica (rigida) dell'imbarcazione

Seconda legge di Newton (**forza = massa × accelerazione**)

$$m\ddot{\mathbf{X}}_G = \mathbf{F} \quad (\text{lineare})$$

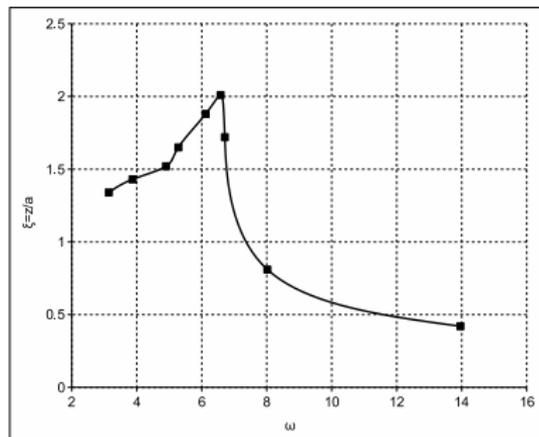
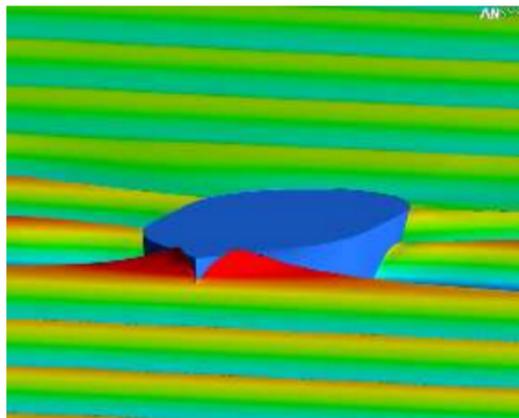
$$\mathcal{R}I_G\mathcal{R}^{-1}\dot{\boldsymbol{\omega}} + \boldsymbol{\omega} \times \mathcal{R}I_G\mathcal{R}^{-1}\boldsymbol{\omega} = \mathbf{M}_G \quad (\text{angolare})$$

# Dinamica (rigida) dell'imbarcazione

Seconda legge di Newton (**forza = massa × accelerazione**)

$$m\ddot{\mathbf{X}}_G = \mathbf{F} \quad (\text{lineare})$$

$$\mathcal{R}I_G\mathcal{R}^{-1}\dot{\boldsymbol{\omega}} + \boldsymbol{\omega} \times \mathcal{R}I_G\mathcal{R}^{-1}\boldsymbol{\omega} = \mathbf{M}_G \quad (\text{angolare})$$



Massimo affondamento vs Frequenza d'onda

# Interazione vento/vele

- Di **bolina**, flusso sostanzialmente attaccato (no separazioni)
- Modelli più semplici di flusso (a potenziale)
- Accoppiamento fluido-struttura tra un solutore strutturale e un codice a pannelli



- Di **poppa**, il flusso separato attorno a spinnaker/gennaker
- Solutori fluidi basati sulle equazioni di Navier-Stokes
- Sviluppo di algoritmi di accoppiamento fluido-struttura

**Solutore strutturale:** data la pressione sulla vela, calcola deformazione e velocità

$$(\mathbf{G}, \mathbf{U}) = \text{Struct}(P)$$

**Solutore fluido:** date posizione e velocità della vela, calcola pressione

$$P = \text{Fluid}(\mathbf{G}, \mathbf{U})$$

**Accoppiamento FSI:** dato un campo iniziale di pressione  $P_0$ , si itera, per  $k = 1, \dots$

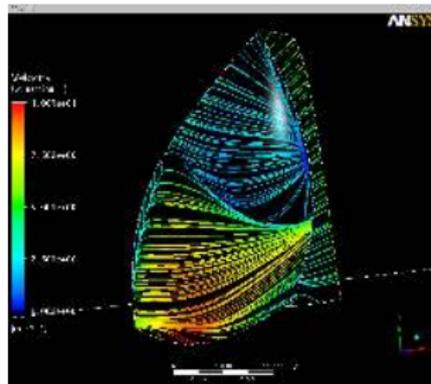
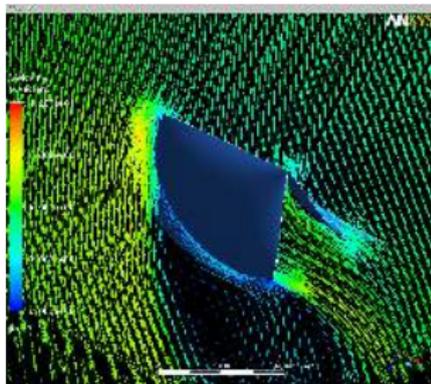
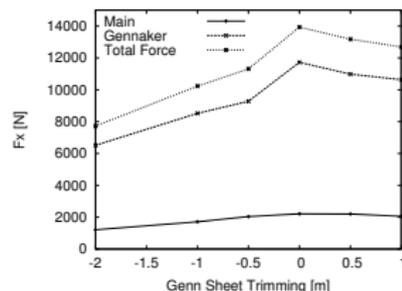
$$(\mathbf{G}_{k+1}, \mathbf{U}_{k+1}) = \text{Struct}(p_k)$$

$$P_{k+1} = \text{Fluid}(\mathbf{G}_{k+1}, \mathbf{U}_{k+1})$$

fino a convergenza.

# Simulazioni FSI stazionarie per le vele

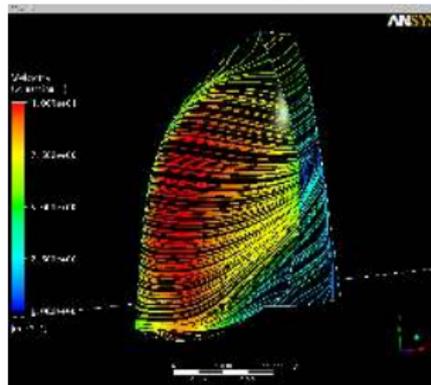
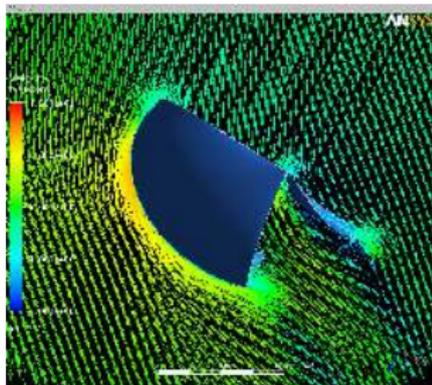
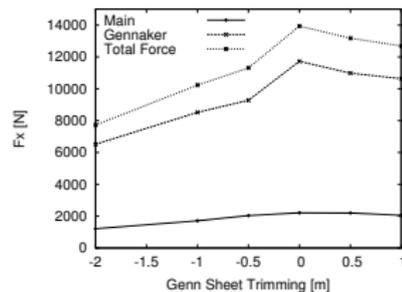
- Analisi di diverse forme e regolazioni (trimming)
- Identificazione del trimming ottimale



Gennaker Sheet Trimming GS=-1 m

# Simulazioni FSI stazionarie per le vele

- Analisi di diverse forme e regolazioni (trimming)
- Identificazione del trimming ottimale



Gennaker Sheet Trimming GS=0.5 m

## 31<sup>a</sup> Coppa America

- Auckland (NZ), Febbraio 2003
- Defender: Team New Zealand (NZ)
- Challenger: Alinghi (SUI)



## 31<sup>a</sup> Coppa America

- Auckland (NZ), Febbraio 2003
- Defender: Team New Zealand (NZ)
- Challenger: Alinghi (SUI)



## 31<sup>a</sup> Coppa America

- Auckland (NZ), Febbraio 2003
- Defender: Team New Zealand (NZ)
- Challenger: Alinghi (SUI)



## 32<sup>a</sup> Coppa America

- Valencia (E), Luglio 2007
- Defender: Alinghi (SUI)
- Challenger: Team New Zealand (TNZ)



## 31<sup>a</sup> Coppa America

- Auckland (NZ), Febbraio 2003
- Defender: Team New Zealand (NZ)
- Challenger: Alinghi (SUI)



## 32<sup>a</sup> Coppa America

- Valencia (E), Luglio 2007
- Defender: Alinghi (SUI)
- Challenger: Team New Zealand (TNZ)



## 31<sup>a</sup> Coppa America

- Auckland (NZ), Febbraio 2003
- Defender: Team New Zealand (NZ)
- Challenger: Alinghi (SUI)



## 32<sup>a</sup> Coppa America

- Valencia (E), Luglio 2007
- Defender: Alinghi (SUI)
- Challenger: Team New Zealand (TNZ)



## 33<sup>a</sup> Coppa America

- Valencia (E), Febbraio 2010
- Defender: Alinghi (SUI)
- Challenger: BMW Oracle Racing (USA)



# Matematica per vincere (purtroppo non sempre)

## 31<sup>a</sup> Coppa America

- Auckland (NZ), Febbraio 2003
- Defender: Team New Zealand (NZ)
- Challenger: Alinghi (SUI)



## 32<sup>a</sup> Coppa America

- Valencia (E), Luglio 2007
- Defender: Alinghi (SUI)
- Challenger: Team New Zealand (TNZ)



## 33<sup>a</sup> Coppa America

- Valencia (E), Febbraio 2010
- Defender: Alinghi (SUI)
- Challenger: BMW Oracle Racing (USA)



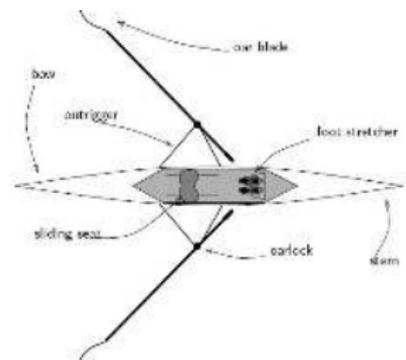
# Matematica e Canottaggio Olimpico

Problema complesso:

- superficie libera
- flussi turbolenti
- dinamica dei vogatori

Modelli sviluppati:

- ricostruzione del sistema scafo/remi/vogatori
- modello dinamico rapido con fluidodinamica semplificata
- accoppiamento con modelli fluidodinamici completi
- analisi dell'effetto della profondità del bacino



Equazioni per la quantità di moto (lineare e angolare) per lo scafo

$$M\ddot{\mathbf{G}}^c = M\mathbf{g} + \mathbf{F}^w$$

$$\mathcal{R}I_G\mathcal{R}^{-1}\dot{\boldsymbol{\omega}} + \boldsymbol{\omega} \times \mathcal{R}I_G\mathcal{R}^{-1}\boldsymbol{\omega} = \mathbf{M}^w$$

# Dinamica dello scafo (con forzanti)

Equazioni per la quantità di moto (lineare e angolare) per lo scafo

$$\begin{aligned} M\ddot{\mathbf{G}}^c &= M\mathbf{g} + \mathbf{F}^w + \sum_{j=1}^n (\mathbf{F}_{olj} + \mathbf{F}_{orj}) + \sum_{j=1}^n (\mathbf{F}_{slj} + \mathbf{F}_{srj}) \\ &+ \sum_{j=1}^n (\mathbf{F}_{flj} + \mathbf{F}_{frj}) \end{aligned}$$

$$\begin{aligned} \mathcal{R}I_G\mathcal{R}^{-1}\dot{\boldsymbol{\omega}} + \boldsymbol{\omega} \times \mathcal{R}I_G\mathcal{R}^{-1}\boldsymbol{\omega} &= \mathbf{M}^w + \sum_{j=1}^n [(\mathbf{X}_{olj} - \mathbf{G}^h) \times \mathbf{F}_{olj} + (\mathbf{X}_{orj} - \mathbf{G}^h) \times \mathbf{F}_{orj}] \\ &+ \sum_{j=1}^n [(\mathbf{X}_{slj} - \mathbf{G}^h) \times \mathbf{F}_{slj} + (\mathbf{X}_{srj} - \mathbf{G}^h) \times \mathbf{F}_{srj}] \\ &+ \sum_{j=1}^n [(\mathbf{X}_{flj} - \mathbf{G}^h) \times \mathbf{F}_{flj} + (\mathbf{X}_{frj} - \mathbf{G}^h) \times \mathbf{F}_{frj}] \end{aligned}$$

$\mathbf{F}_{olj}, \mathbf{F}_{orj}$ : forze sugli scalmi

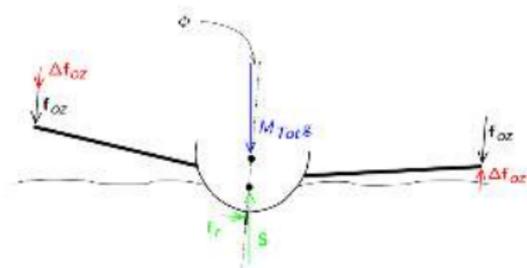
$\mathbf{F}_{slj}, \mathbf{F}_{srj}$ : forze sulle sedute

$\mathbf{F}_{flj}, \mathbf{F}_{frj}$ : forze sulle pedane

# Un semplice modello di controllo

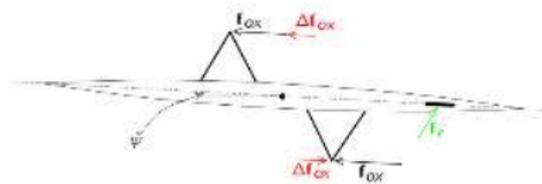
Il sistema è intrinsecamente instabile e richiede un modello di controllo.

## Controllo del rollio



$$F_{o,x} = \begin{cases} F_x^{max} \sin\left(\frac{\pi t}{\tau_a}\right)^2 \pm k_{Roll} \phi, & \text{if } 0 \leq t \leq \tau_a \\ \pm k_{Roll} \phi & \text{if } \tau_a < t \leq T \end{cases}$$

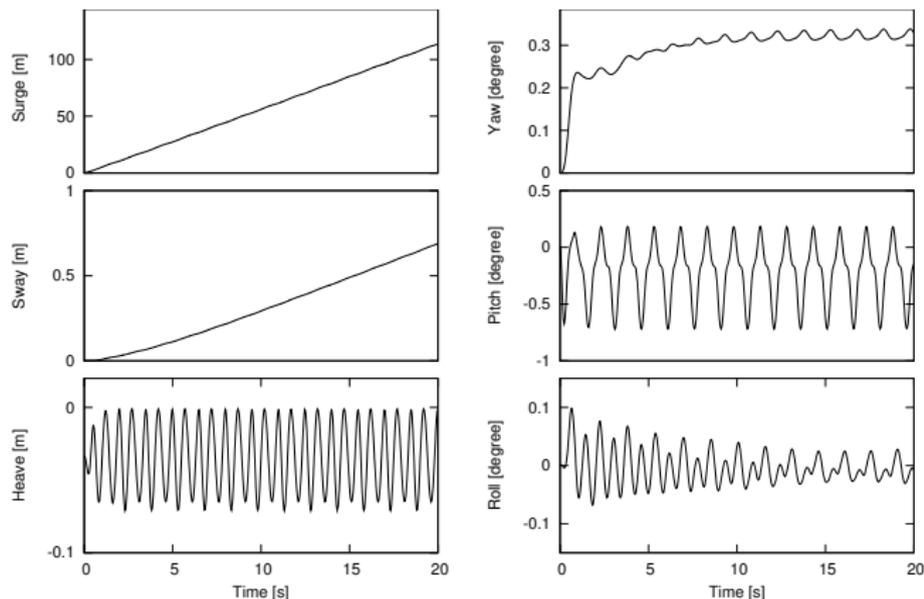
## Controllo dell'imbardata



$$F_{o,z} = \begin{cases} (F_z^{max} \pm k_{Yaw} \psi) \sin\left(\frac{\pi t}{\tau_a}\right)^2, & \text{if } 0 \leq t \leq \tau_a \\ 0 & \text{if } \tau_a < t \leq T \end{cases}$$

# Simulazioni di regate

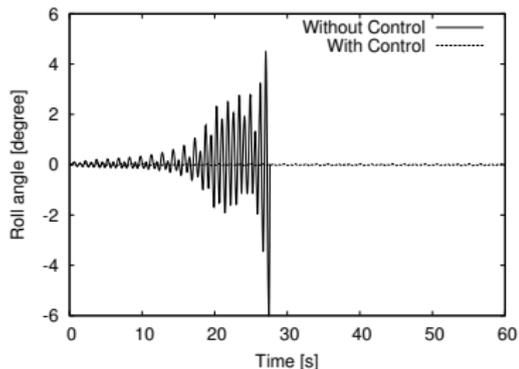
## Caso non-simmetrico: 4 senza (di punta)



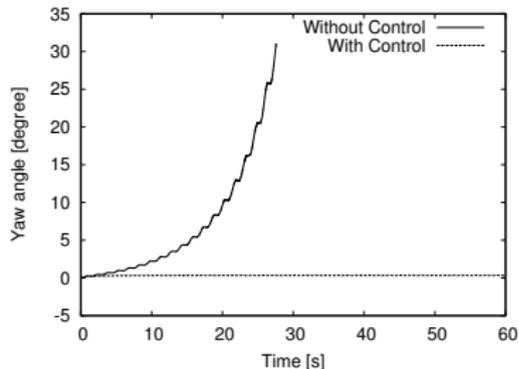
$$\begin{array}{ll} F_{X_{max} 1} = 1300N & F_{Z_{max} 1} = 230N \\ F_{X_{max} 2} = 1200N & F_{Z_{max} 1} = 200N \\ F_{X_{max} 3} = 1200N & F_{Z_{max} 1} = 200N \\ F_{X_{max} 4} = 1200N & F_{Z_{max} 1} = 200N \end{array}$$

# Efficacia del controllo

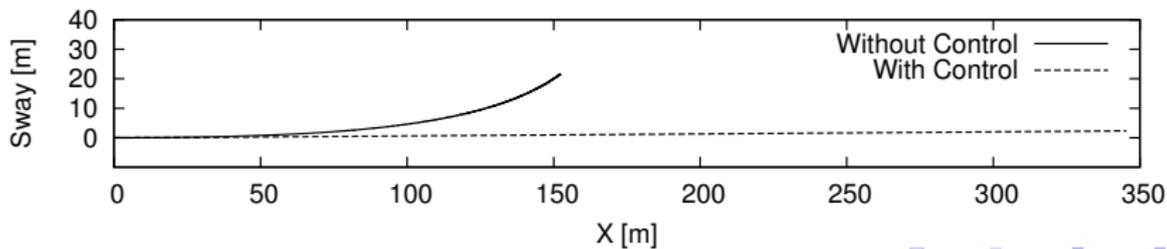
## Rollio



## Imbardata

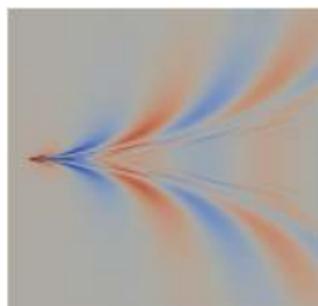


## Traiettoria dell'imbarcazione



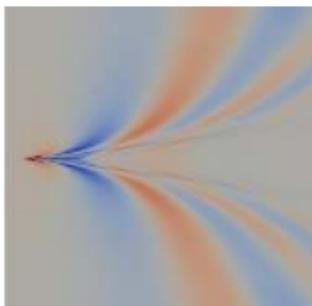
# Profondità del bacino di gara

- Ricerca commissionata della Federazione internazionale (FISA)
- Quanto influisce la profondità del bacino sulle prestazioni
  - diverso comportamento delle correnti nel bacino
  - diversi regimi (acque alte o basse)
  - analisi dei profili d'onda a diversi regimi
  - analisi dell'interferenza tra le corsie



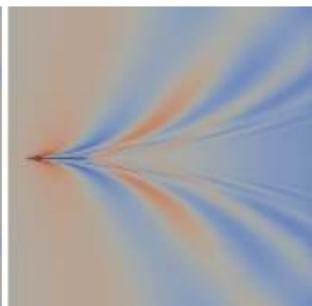
H=2 m

$Fr_H = 0.85$



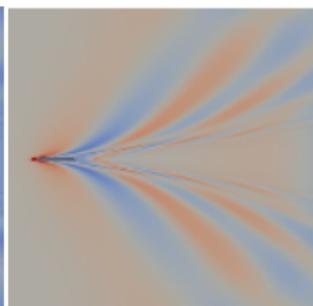
H=2.5 m

$Fr_H = 0.92$



H=3 m

$Fr_H = 1.00$



H=3.5 m

$Fr_H = 1.12$

# Modelli numerici per il nuoto



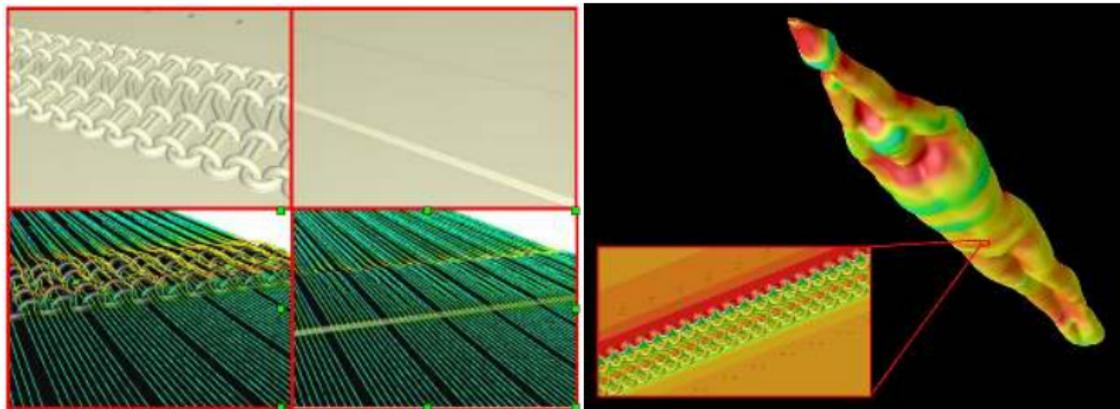
- Da qualche anno, grande sviluppo della tecnologia per i costumi da competizione
- Miglioramento delle prestazioni dovuto a nuovi materiali, tecniche di assemblaggio, analisi sperimentali e numeriche
- Dal 2009, nuovo regolamento (molto restrittivo) della FINA

## Collaborazione con Arena

- **2004** - Sviluppo del modello Powerskin Extreme e analisi del comportamento fluidodinamico del nuovo tessuto (A. Veneziani, E. Foa)
- **2008** - Analisi del modello Powerskin R-Evolution per la quantificazione del potenziale guadagno associato alla nuova tecnologia di assemblaggio

# Effetto delle cuciture sulla resistenza

- Cuciture rimosse o sostituite con incollaggi (piatti)
- Miglioramento delle prestazioni dovuto alla minore resistenza
- Simulazioni numeriche **locali** (nella regione delle cuciture) e **globali** per la stima della riduzione di resistenza



Cucitura	Drag variation		
	1.8 m/s	2.0 m/s	2.3 m/s
spalle, sost.	0.056 N (0.14%)	0.064 N (0.12%)	0.102 N (0.15%)
schiena, rim.	0.056 N (0.14%)	0.071 N (0.14%)	0.102 N (0.15%)
vita, rim.	0.099 N (0.24%)	0.128 N (0.24%)	0.180 N (0.27%)
inguine, rim.	0.037 N (0.09%)	0.048 N (0.09%)	0.066 N (0.10%)
caviglia, sost..	0.031 N (0.08%)	0.035 N (0.07%)	0.050 N (0.07%)
<b>Totale</b>	<b>0.279 N (0.68%)</b>	<b>0.346 N (0.66%)</b>	<b>0.499 N (0.74%)</b>

Riduzione di resistenza

	Scivolamento [s]	Vasca [s]	Gara [s]
50 m	0.011	0.073	0.073
100 m	0.011	0.077	0.154
200 m	0.012	0.083	0.332
400 m	0.013	0.087	0.696

Miglioramenti sul tempo di gara

- I miglioramenti sul tempo di gara sono calcolati risolvendo una semplice equazione differenziale che descrive la dinamica del nuotatore.

# Fare una vasca con un'equazione differenziale

$$\begin{cases} m \frac{d^2 x(t)}{dt^2} = F(t) \\ v(t)|_{t=0} = \frac{dx}{dt}|_{t=0} = 0 \\ x|_{t=0} = 0 \end{cases}$$

# Fare una vasca con un'equazione differenziale

$$\begin{cases} m \frac{d^2 x(t)}{dt^2} = F(t) & F(t) = P(t) - D(t) \\ v(t)|_{t=0} = \frac{dx}{dt}|_{t=0} = 0 \\ x|_{t=0} = 0 \end{cases}$$

# Fare una vasca con un'equazione differenziale

$$\left\{ \begin{array}{l} m \frac{d^2 x(t)}{dt^2} = F(t) \\ v(t)|_{t=0} = \frac{dx}{dt}|_{t=0} = 0 \\ x|_{t=0} = 0 \end{array} \right.$$

$$F(t) = P(t) - D(t)$$

$$P(t) = A \sin(2\pi ft) \quad (\text{propulsione})$$

# Fare una vasca con un'equazione differenziale

$$\left\{ \begin{array}{l} m \frac{d^2 x(t)}{dt^2} = F(t) \\ v(t)|_{t=0} = \frac{dx}{dt}|_{t=0} = 0 \\ x|_{t=0} = 0 \end{array} \right.$$

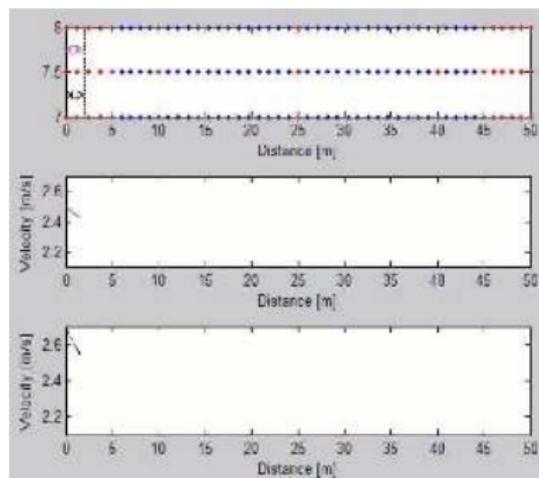
$$F(t) = P(t) - D(t)$$

$$P(t) = A \sin(2\pi ft) \quad (\text{propulsione})$$

$$D(t) = \frac{1}{2} \rho v(t)^2 S C_D \quad (\text{resistenza})$$

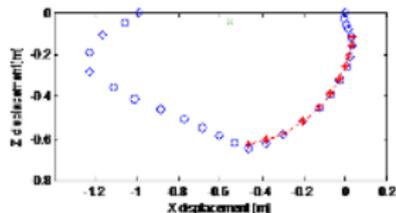
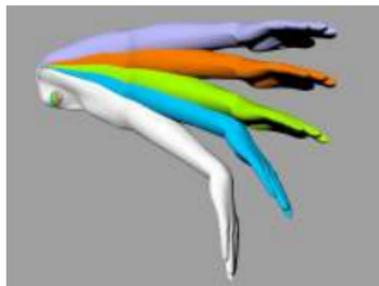
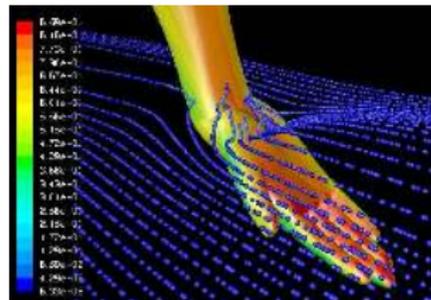
# Fare una vasca con un'equazione differenziale

$$\left\{ \begin{array}{l} m \frac{d^2 x(t)}{dt^2} = F(t) \\ v(t)|_{t=0} = \frac{dx}{dt}|_{t=0} = 0 \\ x|_{t=0} = 0 \end{array} \right. \quad \begin{array}{l} F(t) = P(t) - D(t) \\ P(t) = A \sin(2\pi ft) \quad (\text{propulsione}) \\ D(t) = \frac{1}{2} \rho v(t)^2 S C_D \quad (\text{resistenza}) \end{array}$$

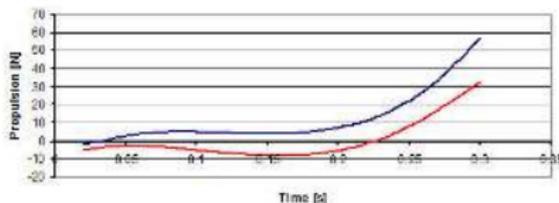
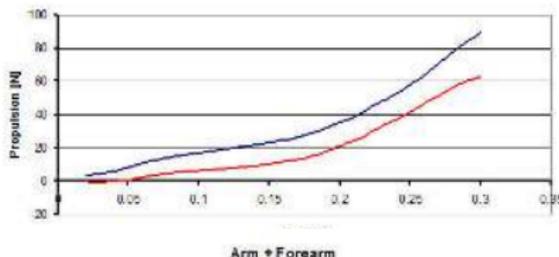


# Verso simulazioni più complesse: la bracciata

- Simulazione CFD della dinamica della bracciata
- Ricostruzione cinematica della bracciata
- Modello di rotazione a singola rotazione (attorno alla spalla) e doppia rotazione (spalla+gomito)



Traiettoria della mano

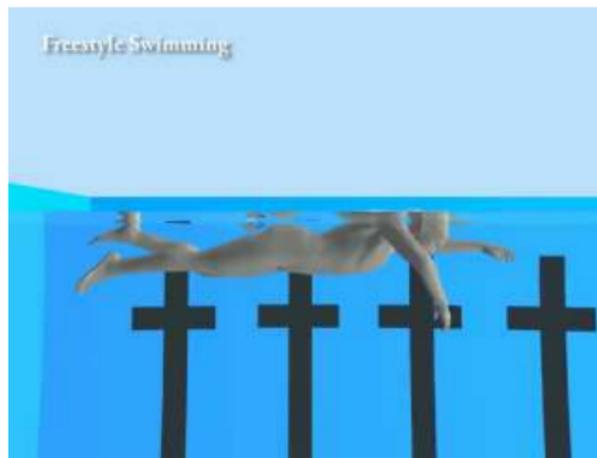


# CFD nel nuoto: dove siamo oggi

Authors	Participants	Object	2D/3D	Treatment of motion	CFD code	Turbulence model*
Kawai (1997)	Dolphin kick	Leg and foot	2D	Cut cell	STREAM@	
Sato and Hino (2003, 2010)	Stroke of swimmer	Hand	3D	Moving grid	SURF	Without
Rouboa et al. (2006)	Linear acceleration	Hand and forearm	3D	Body force	Fluent@	$k-\epsilon$
Lecrivain et al. (2008)	Stroke of upper arm	Body and upper arm	3D	Dynamic mesh	Fluent@	N/A
von Loebbecke et al. (2009b, 2009c)	Dolphin kick	Full body	3D	IB	VICAR3D	Without
Keys (2010)	Dolphin, freestyle and breaststroke kick Freestyle swimming at water surface	Full body	3D		Fluent@	Realisable $k-\epsilon$
von Loebbecke and Mittal (2012)	Stroke of swimmer	Hand and arm	3D	IB	VICAR3D	Without
Cohen et al. (2012)	Dolphin kick	Full body	3D	SPH	In-house	Without
Hochstein et al. (2012)	Dolphin kick	Full body	3D		OpenFOAM	N/A
Sato and Hino (2013)	Stroke of swimmer	Hand	3D	Moving grid	SURF	Without
Pacholak et al. (2014)	Dolphin kick	Full body	3D		OpenFOAM	N/A

# CFD nel nuoto: dove siamo oggi

Authors	Participants	Object	2D/3D	Treatment of motion	CFD code	Turbulence model*
Kawai (1997)	Dolphin kick	Leg and foot	2D	Cut cell	STREAM®	
Sato and Hino (2003, 2010)	Stroke of swimmer	Hand	3D	Moving grid	SURF	Without
Rouboa et al. (2006)	Linear acceleration	Hand and forearm	3D	Body force	Fluent®	$k-\epsilon$
Lecrivain et al. (2008)	Stroke of upper arm	Body and upper arm	3D	Dynamic mesh	Fluent®	N/A
von Loebbecke et al. (2009b, 2009c)	Dolphin kick	Full body	3D	IB	VICAR3D	Without
Keys (2010)	Dolphin, freestyle and breaststroke kick Freestyle swimming at water surface	Full body	3D		Fluent®	Realisable $k-\epsilon$
von Loebbecke and Mittal (2012)	Stroke of swimmer	Hand and arm	3D	IB	VICAR3D	Without
Cohen et al. (2012)	Dolphin kick	Full body	3D	SPH	In-house	Without
Hochstein et al. (2012)	Dolphin kick	Full body	3D		OpenFOAM	N/A
Sato and Hino (2013)	Stroke of swimmer	Hand	3D	Moving grid	SURF	Without
Pacholak et al. (2014)	Dolphin kick	Full body	3D		OpenFOAM	N/A



KAIST Motion Computing Laboratory

# L'importanza dei dati

## Analisi tecnico-tattica

- Grande mole di dati raccolti (posizioni, passaggi, ...)
- La sfida estrarre informazioni utili (anche in tempo reale)
- Strumenti statistici innovativi per l'analisi dei dati e delle reti
- Sviluppo di un Virtual Coach a disposizione dei tecnici



## Analisi tecnico-tattica

- Grande mole di dati raccolti (posizioni, passaggi, ...)
- La sfida estrarre informazioni utili (anche in tempo reale)
- Strumenti statistici innovativi per l'analisi dei dati e delle reti
- Sviluppo di un Virtual Coach a disposizione dei tecnici



## Analisi tecnico-tattica

- Grande mole di dati raccolti (posizioni, passaggi, ...)
- La sfida estrarre informazioni utili (anche in tempo reale)
- Strumenti statistici innovativi per l'analisi dei dati e delle reti
- Sviluppo di un Virtual Coach a disposizione dei tecnici



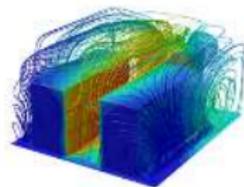
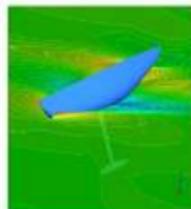
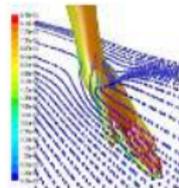
## Analisi del gesto

- Sviluppo di modelli numerici e statistici per lo sport (volley, golf, tennis)
- MOVIDA: tracking e analisi del gesto con strumenti di statistica funzionale e machine learning
- Collaborazione con CONI

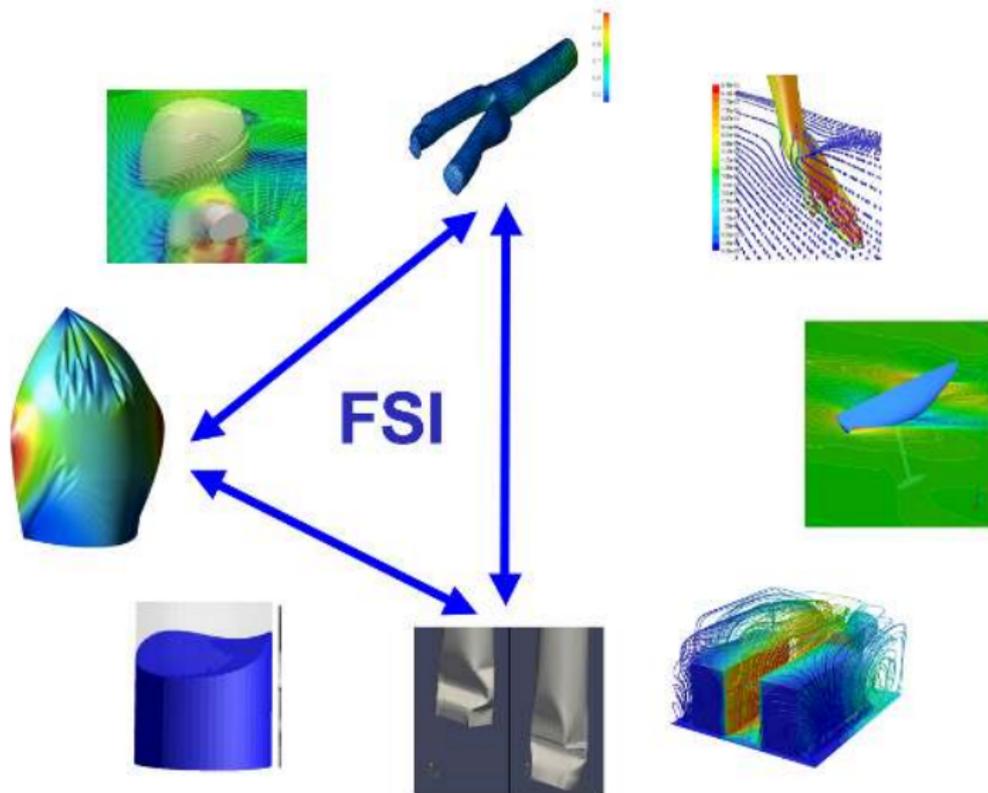


Attività svolta dalla start-up **MathAndSport**

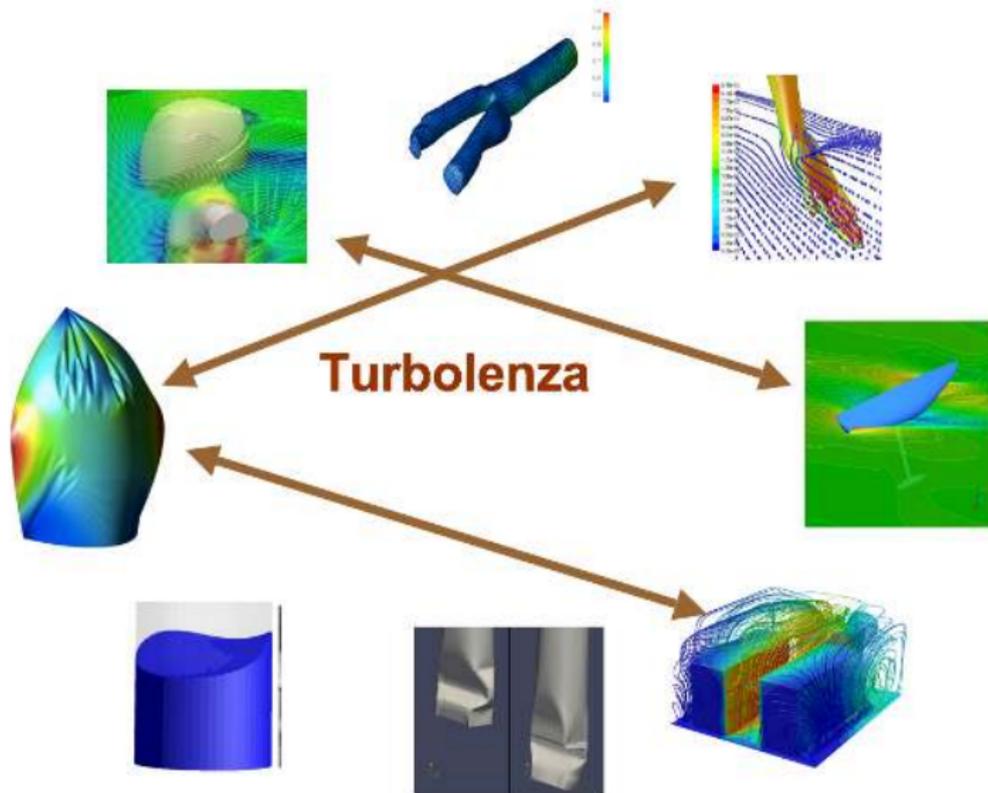
# Sport: solo uno dei tanti campi da gioco



# Sport: solo uno dei tanti campi da gioco

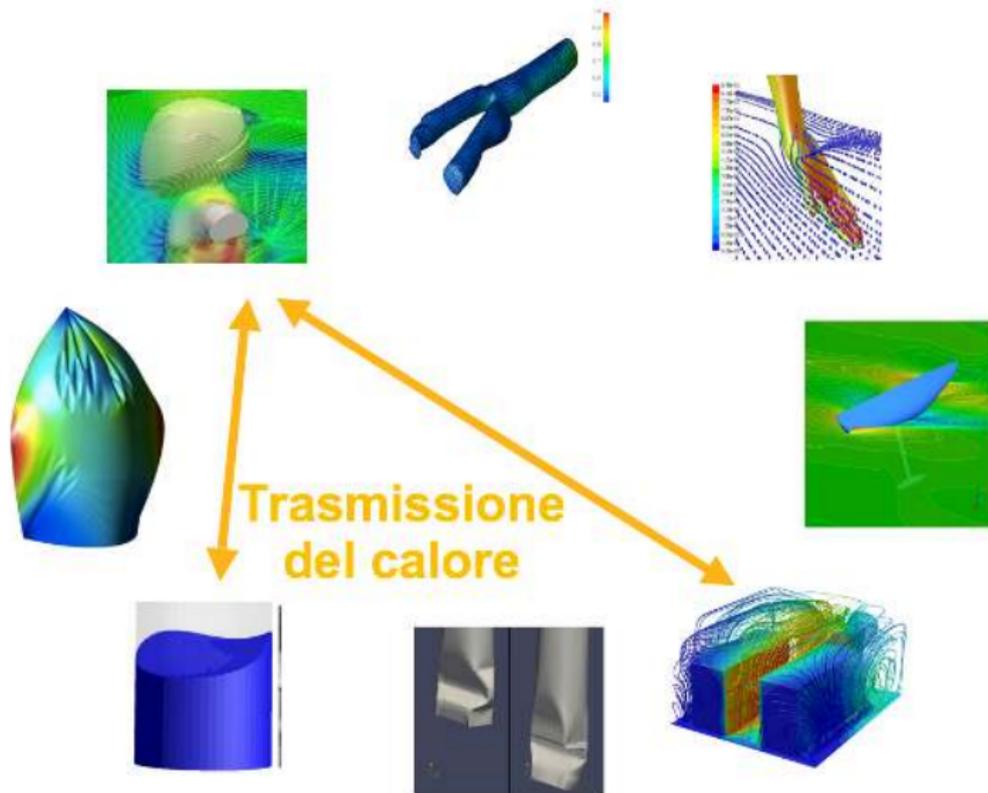


# Sport: solo uno dei tanti campi da gioco

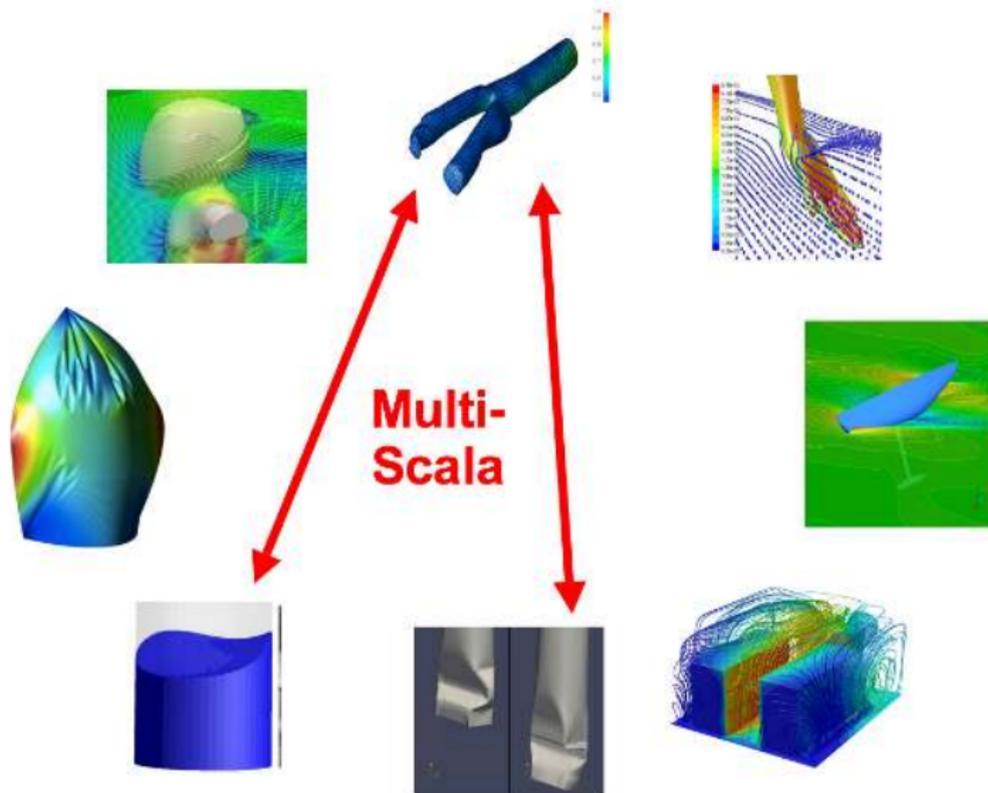




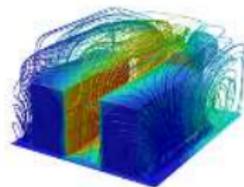
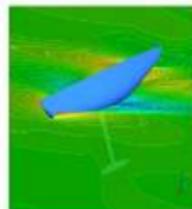
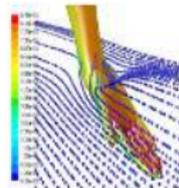
# Sport: solo uno dei tanti campi da gioco



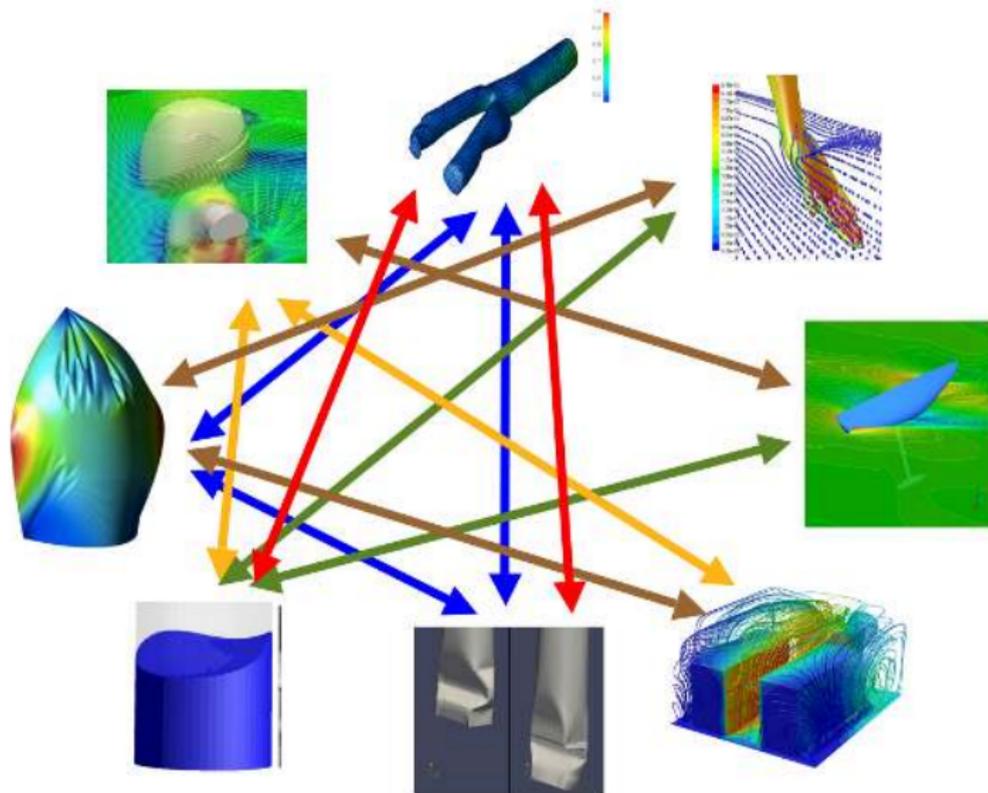
# Sport: solo uno dei tanti campi da gioco



# Sport: solo uno dei tanti campi da gioco



# Sport: solo uno dei tanti campi da gioco



GRAZIE PER  
L'ATTENZIONE