

Capitolo 4

Specifica funzionale e implementazione

4.1 Introduzione

Nel capitolo precedente è stata illustrata l'architettura funzionale del progetto, specificando dove necessario, l'interazione tra utente ed applicazione, ed indicando alcuni dei punti critici da considerare in fase implementativa.

Nel presente capitolo viene illustrato la corrispondenza tra la specifica funzionale e l'implementazione, giustificando le scelte operate in termini di algoritmi e di GUI.

La GUI dell'applicazione è stata realizzata con l'ausilio dell'ambiente GUI MATLAB 6.5 arricchito grazie all'uso di ACTIVEX importati con le funzionalità '*actxcontrol*' e '*actxserver*' delle quali si dirà ampiamente in seguito e nell'APPENDICE A.

La scelta di utilizzare l'ambiente di sviluppo GUI di MATLAB è maturata dall'esame delle funzionalità dell'applicazione e dalla possibilità di farla interagire direttamente con i toolbox di simulazione SIMULINK® e STATEFLOW®. Tale scelta ha comportato alcuni svantaggi come la difficoltà nella compilazione.

4.2 Avvio dell'applicazione

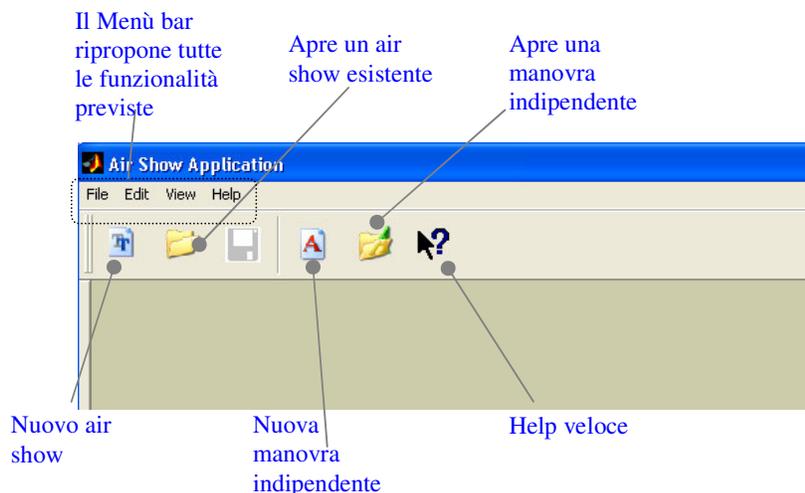
L'applicazione può essere lanciata digitando il comando *Aresti* dal prompt di MATLAB 6.5 (*per la versione matlab*), o nel caso si sia utilizzata la procedura di installazione, (*APPENDICE C*) lanciando l'eseguibile (*di cui è possibile creare un collegamento sul desktop*), che non fa altro che creare un'istanza del Matlab Server al quale vengono inviati i comandi.

La versione eseguibile dell'applicazione può essere lanciata senza che ci siano errori run-time solo se sono state preventivamente installate tutte le librerie matlab utilizzate.

All'avvio, l'interfaccia dell'applicazione permette solo le seguenti macro funzionalità

- Crea nuovo air show
- Open air show
- New aresti symbol
- Open aresti symbol
- Help veloce

Figura 4.1



L'help veloce è utile per l'utente che usa per la prima volta l'applicazione. Serve a dare una spiegazione veloce sulla funzionalità del comando selezionato. La funzionalità di help veloce è prevista per i comandi successivi alla creazione o caricamento di un air show o di una

manovra predefinita. L'utente sa che tale funzionalità è abilitata se il puntatore del mouse si presenta come un ?
 Di seguito il codice aggiunto alla funzione 'nomhandle_callback (...)' dove *nomhandle* è il nome (campo Tag) associato all'handle selezionato con il mouse.

```

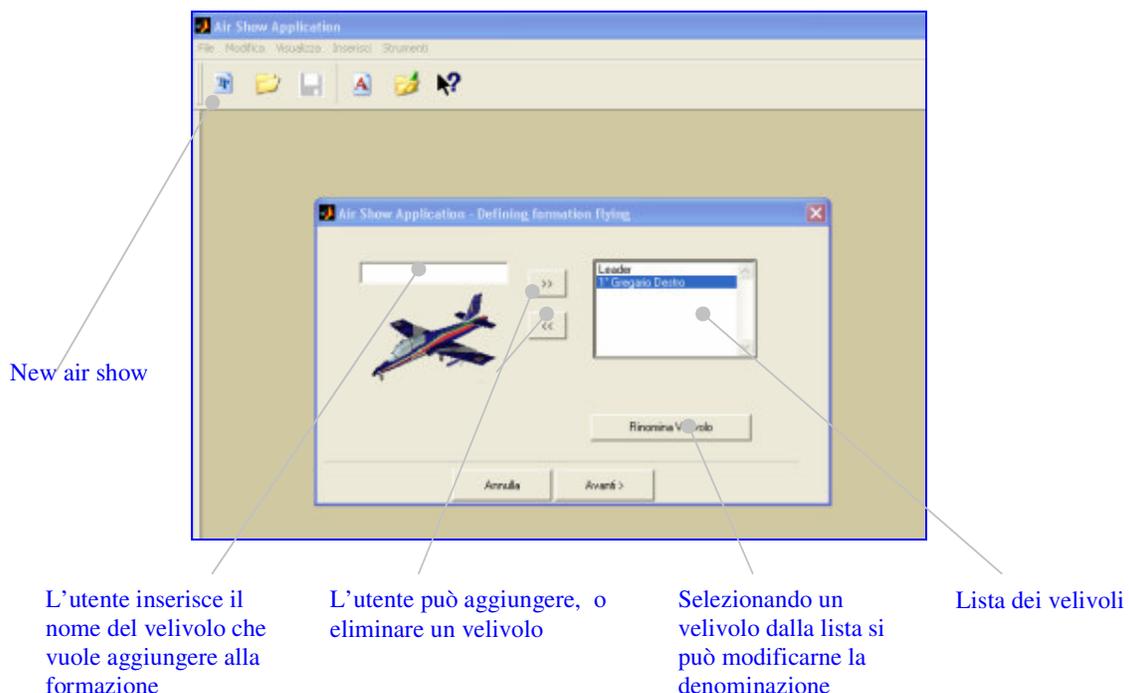
if getappdata(handles.Main,'tipopointer') == 1
    str = '.....'; % stringa help visualizzata nella parte bassa del form
    set(handles.information,'String',str);
else
    % funzionalità associata al comando.....
end
    
```

- 'tipopointer' è una variabile GUI a cui viene assegnato un valore dipendente dal tipo di puntatore mouse attivo.
- 'information' è il nome di un oggetto static text il cui campo stringa è atto a contenere la stringa di aiuto per la funzionalità.

4.3 Creare o aprire un air show

Come detto nella specifica ad alto livello la definizione della formazione è necessaria alla creazione di un nuovo air show.

Figura 4.2



New air show

L'utente inserisce il nome del velivolo che vuole aggiungere alla formazione

L'utente può aggiungere, o eliminare un velivolo

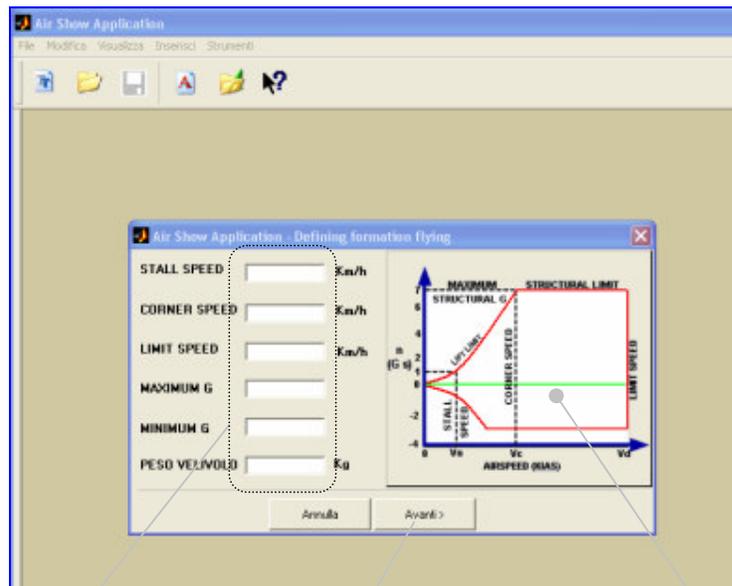
Selezionando un velivolo dalla lista si può modificarne la denominazione

Lista dei velivoli

La denominazione dei velivoli è decisa in questa fase e non è più modificabile, per cui ho ritenuto opportuno prevedere la possibilità (*es. immissione errata*) di modificare i nomi dei velivoli la cui lista può facilmente essere esaminata e modificata (fig. 4.2).

Il passo successivo è costituito dalla definizione dei parametri del velivolo.

Figura 4.3



Parametri dell'involuppo di volo.
In caso di inserimento di valori non numerici viene data indicazione dell'errore

L'utente può proseguire solo dopo aver settato tutti i parametri dell'involuppo di volo

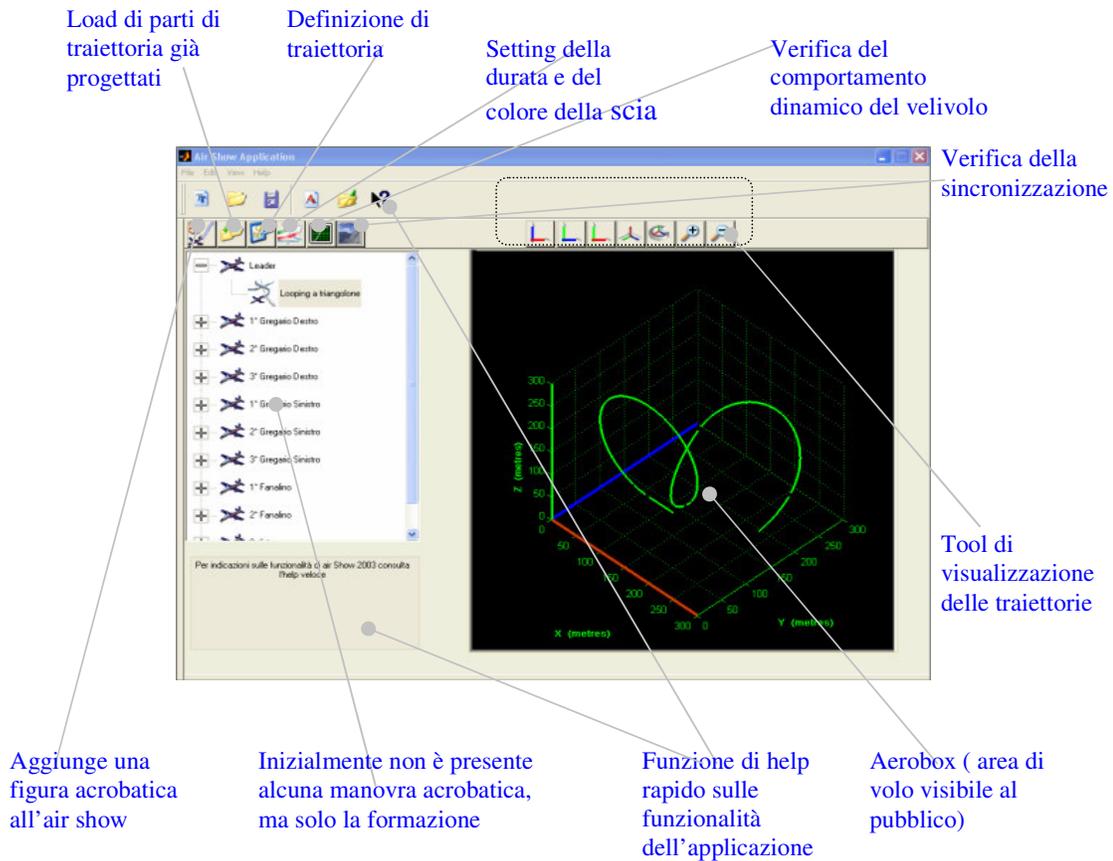
Rappresentazione di un tipico diagramma di manovra per velivoli acrobatici

Di fianco ai parametri (fig. 4.3) è riportato un tipico diagramma di manovra utile ad avere una corrispondenza diretta tra i campi da compilare e il loro significato fisico.

La definizione dei parametri dell'involuppo di volo è fondamentale, per cui l'utente dovrà fare particolare attenzione e controllare bene i valori introdotti e le unità di misura di riferimento.

Al passo successivo vengono aggiornate le strutture dati e l'oggetto *image control tree* (fig. 4.4).

Figura 4.4



L'oggetto *image control tree* è stato importato con il comando

```
[j,f]=gcbo
hTree=actxcontrol('MSComctlLib.TreeCtrl.2',[10 190 300 400],f,{'Click'
....'allevents'; 'DbtClick' 'allevents'; 'MouseDown' 'allevents'});
```

Le azioni di interazione con l'oggetto da parte dell'utente sono gestite da una funzione esterna alla GUI *allevents.m*¹. Di seguito un esempio di aggiornamento dei nodi costituenti l'oggetto.

```
hNodeCollection = get(hTree,'Nodes');
h=invoke(hNodeCollection,'Add');
set(h,'Text',Velivoli{i});
set(h,'Image',1);
```

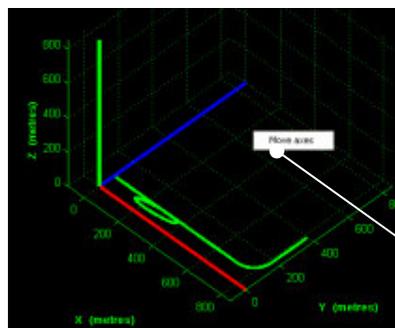
Dalla figura 4.4 si può notare la disposizione dei bottoni relativi alla funzionalità di definizione dell'air show e immediatamente sopra l'oggetto *image control tree*. Con lo stesso criterio sono stati disposti i bottoni utili alla visualizzazione delle traiettorie.

Si è dato particolare importanza all'esame visivo delle traiettorie, infatti oltre alle comuni funzionalità di

- cambio vista
- rotazione dell'aerobox
- zoom

ho previsto che l'utente possa spostare l'oggetto aerobox per poter sfruttare al massimo lo zoom anche su zone ben definite della traiettoria.

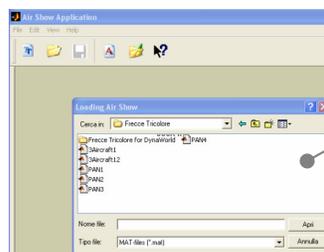
Figura 4.5



Attiva la funzione che permette di spostare l'oggetto che viene fissato cliccando nel punto desiderato

Per i particolari su questa funzionalità si rimanda all'appendice A dove è riportato il codice da aggiungere alla GUI.

Per caricare un air show esistente l'utente dovrà scegliere il nome del file in cui è stato memorizzato.



Browser per la scelta del file .mat in cui è salvato l'air show

Figura 4.6

Poiché la struttura AirShow è salvata in un file .mat (*matlab data file*) è necessario che, nel caso non sia presente tale struttura, venga segnalato un messaggio d'errore.

4.4 Aggiungere una figura acrobatica

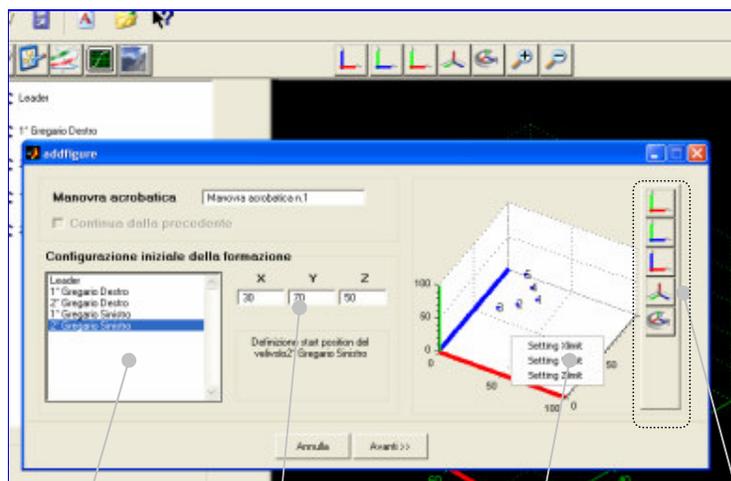
Aggiungere una nuova figura acrobatica comporta la definizione della posizione iniziale della formazione o di un gruppo di velivoli e la denominazione della figura stessa.

La definizione della prima figura acrobatica non permette la possibilità di proseguire la manovra precedente essendo la prima.

L'utente aggiungendo una manovra acrobatica:

- ha la possibilità di impostare la posizione iniziale dei velivoli
- può verificare visivamente la posizione iniziale dei velivoli

Figura 4.7



Lista dei velivoli della formazione

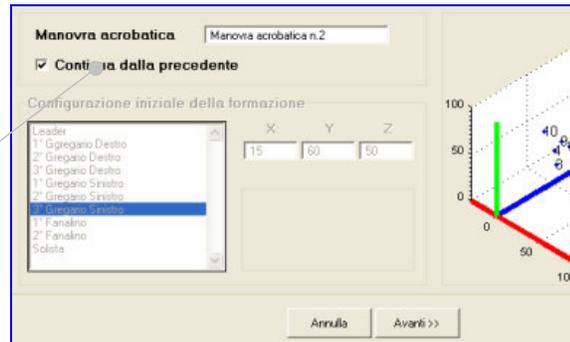
Viene specificata la posizione iniziale dei velivoli

E' possibile settare i limiti visibili dell'aerobox nel caso la formazione non sia visibile al pubblico ad esempio nelle manovre di raccordo.

Tool di visualizzazione per valutare la posizione iniziale dei velivoli

Figura 4.8

L'utente può decidere di far coincidere la posizione iniziale dei velivoli con quella finale della manovra precedente ad esempio nelle manovre di raccordo



L'utente può decidere di assegnare come posizione iniziale alla formazione quella finale della manovra precedente, nel caso ad esempio si voglia progettare una manovra di raccordo.

4.5 Definire una manovra

L'interfaccia per la definizione di una traiettoria permette all'utente:

- Di avere a disposizione gli oggetti base di definizione

Line block 

Circle block 

Follow block 

- Di avere una visione d'insieme dei parametri definiti fino a quel momento
 - Roll
 - Speed
 - Trace

Prevedendo la definizione di traiettorie abbastanza lunghe ho ritenuto necessario poter visionare tali parametri su tratti più brevi di traiettoria (*vedi slider fig 4.8*)

- Di visualizzare la composizione della sequenza definita (*oggetto strip block*)
- Di rapportare la traiettoria definita a quella degli altri velivoli o di uno solo di essi (*fig 4.9*)
 - Questa funzionalità permette di avere un confronto immediato con lo stato di definizione delle traiettorie degli altri velivoli.
- Di visualizzare i particolari di alcuni tratti di essa (*tool di visualizzazione*)

Figura 4.9

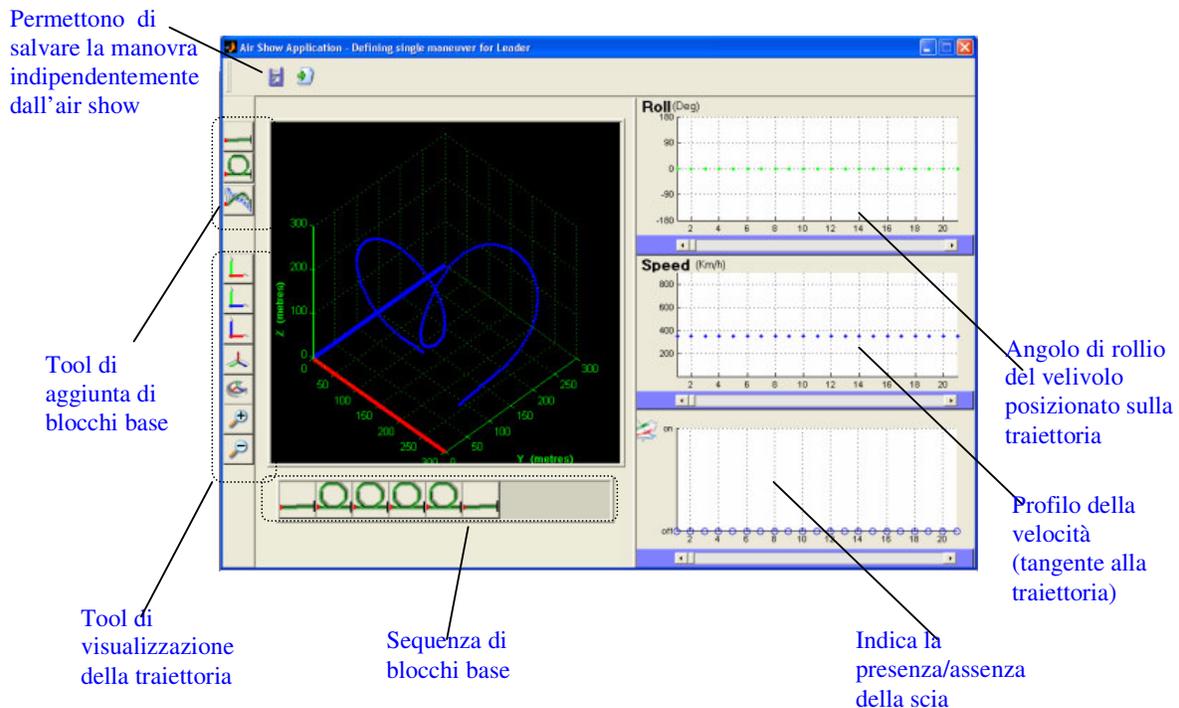
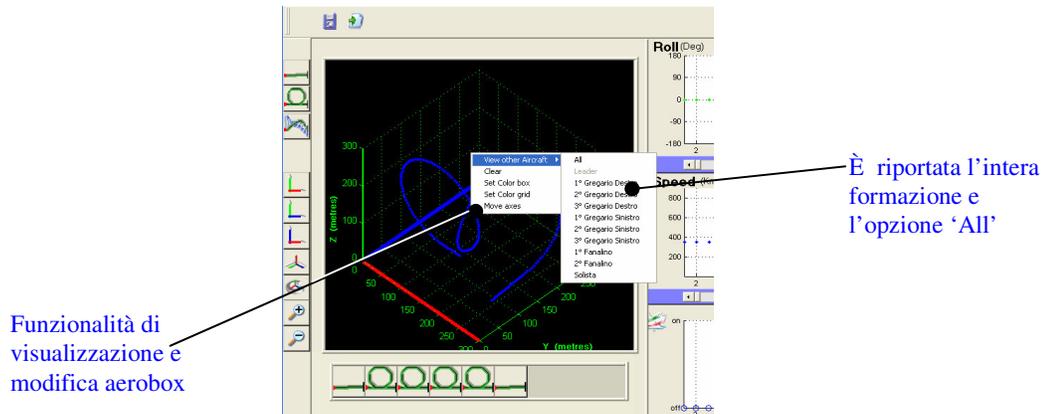


Figura 4.10



Per quanto riguarda le funzionalità di visualizzazione e modifica dell'aerobox, sono distinte in due categorie:

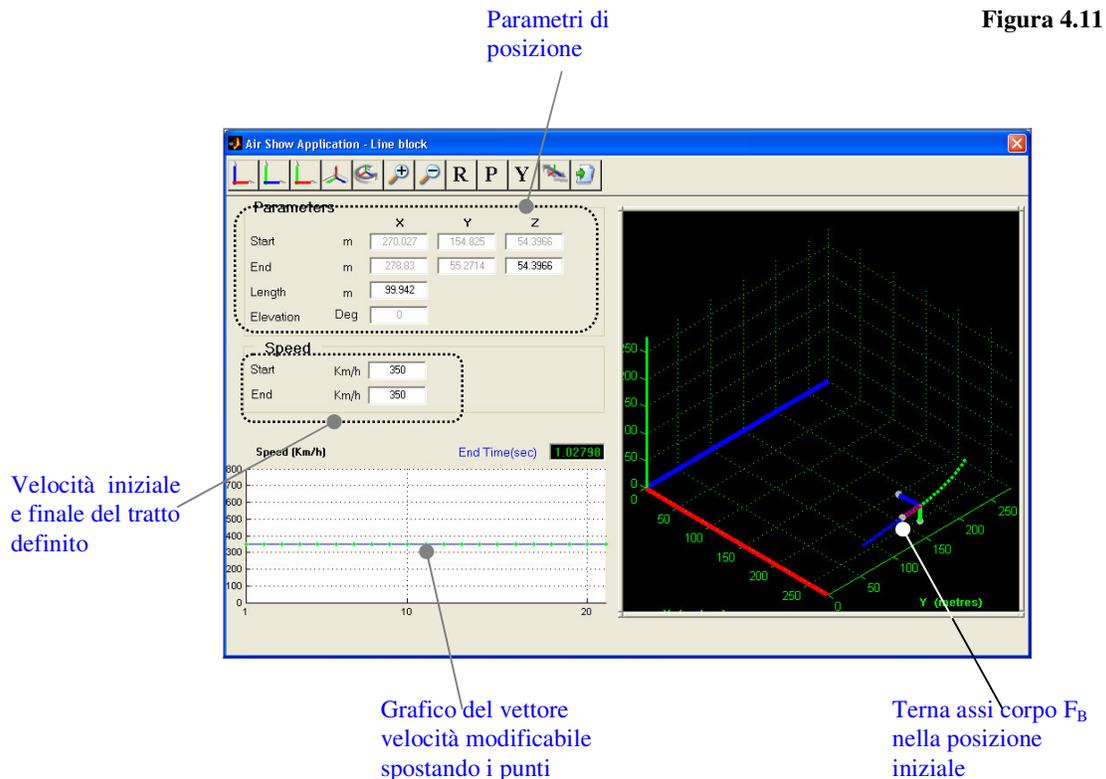
- Visualizzazione delle traiettorie definite
- Proprietà dell'oggetto aerobox
 - Clear (per eliminare le traiettorie relative ad altri velivoli)
 - Color axes (*permette di personalizzare il colore dell'aerobox*)
 - Color grid (*permette di personalizzare il colore della griglia*)
 - Move axes (*permette di spostare l'oggetto con riferimento relativo al punto indicato dall' utente*) è una funzionalità utile ad una migliore visibilità delle traiettorie o di parti di esse.

Le proprietà *Color axes* e *Color grid* sono mantenute permanenti anche nelle successive sessioni di definizione (*le impostazioni sono salvate nel file 'Colorset.mat'*).

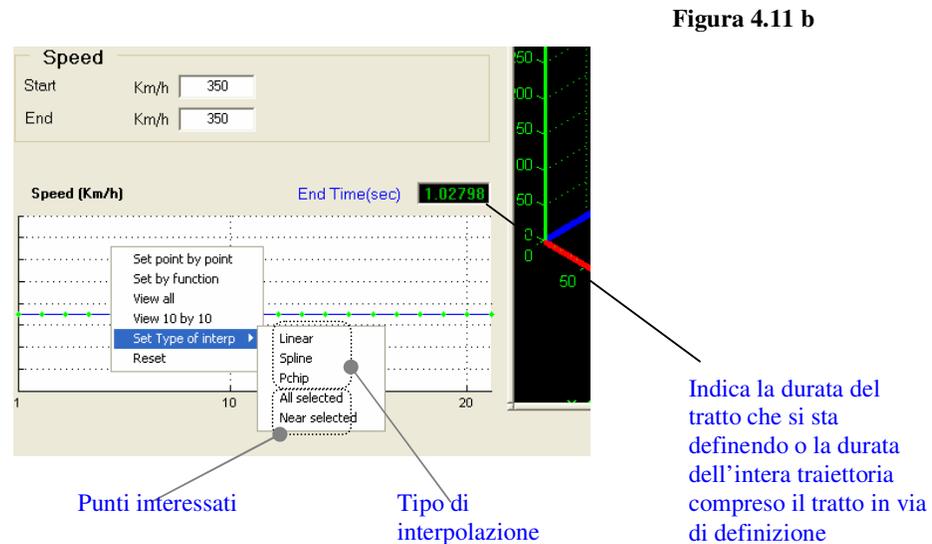
4.5.1 Aggiungere un line block alla sequenza

La sequenza di definizione può iniziare con un *line block* i cui parametri di definizione si distinguono in:

- Parametri di posizione dell'oggetto
 - Posizione iniziale secondo il riferimento inerziale (§ 2.4)
 - Posizione finale secondo il riferimento inerziale (§ 2.4)
 - Lunghezza
 - Angolo di salita (*opzionale rispetto alla posizione iniziale e finale*)
- Vettore velocità di volo (*orientato secondo l'asse X_B della terna assicorpo*) 
- Vettore angolo di rollio R



Il vettore velocità di volo e il vettore angolo di rollio sono settati sfruttando la funzionalità *drag and drop* e tenendo conto dei vincoli dell'inviluppo di volo (§ 2.3).



Il punto selezionato assume i valori puntati dal mouse sul riferimento cartesiano influenzando il valore degli altri punti a seconda del tipo di interpolazione

- Lineare
- Spline
- Pchip

e dei punti interessati

- Tutti quelli interessati dalle precedenti modifiche
- Solo quelli interessati dalle precedenti modifiche con restrizione a quello che precede e segue il punto selezionato

Le variazioni imposte al vettore velocità modificano, così come quelle imposte alla lunghezza, la durata del tratto che si sta definendo (fig. 4.1b)

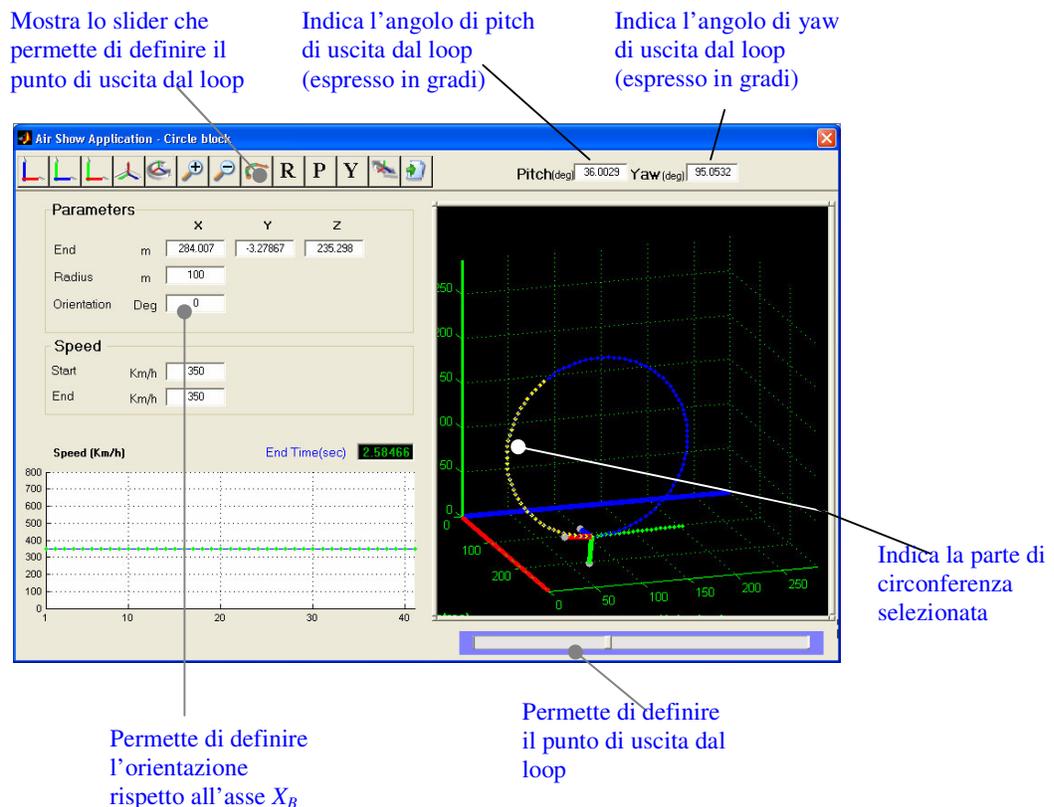
L'utente avrà cura di settare prima, l'andamento del vettore angolo di rollio e successivamente il vettore velocità, affinché la procedura assistita che vincola i valori a quelli di volo ammissibile siano corretti (§2.2).

4.5.2 Aggiungere un circle block alla sequenza

La sequenza di definizione non può iniziare con un *circle block* i cui parametri di definizione si distinguono in:

- Parametri di posizione dell'oggetto
 - Posizione finale secondo il riferimento inerziale (§ 2.6) *(quella iniziale è imposta dal tratto precedente)*
 - Raggio
 - Orientazione *(indica la rotazione dell'oggetto circle intorno all'asse X_B)*
- Vettore velocità di volo *(orientato secondo l'asse X_B della terna assicurata)* 
- Vettore angolo di rollio R

Figura 4.12



Eseguendo il prodotto vettoriale tra i vettori v_y e v_x ricavo il vettore perpendicolare al piano in cui giace la traiettoria Y_B , ed eseguendo il prodotto vettoriale tra v_x e il nuovo v_y viene calcolato Z_B , mentre X_B è il vettore v_x normalizzato. Conoscendo l'orientazione dei vettori Z_B' e Z_B calcolo l'angolo Roll di cui deve essere ruotata la terna F_B' affinché coincida con la terna F_B . Il vettore Y_B è sufficiente calcolarlo una sola volta.

4.5.3 Aggiungere un follow block alla sequenza

Aggiungere alla sequenza di definizione di una traiettoria un *follow block* permette :

- Di inseguire l'evolvere della terna assi corpo del leader (*uno qualsiasi dei velivoli per il quale sia stata definita la traiettoria*)
- La sincronizzazione implicita tra il tratto di traiettoria del leader e quello del wingman

I parametri di definizione per un follow block sono i seguenti:

- Parametri di posizione relativa
 - Intervalli di definizione delle funzioni di posizione
 - $\Delta x(k)$ funzione posizione, relativa dell'origine della terna assi corpo F_B^W (*wingman*) lungo il vettore X_B^L (*leader*)
 - $\Delta y(k)$ funzione posizione, relativa dell'origine della terna assi corpo F_B^W (*wingman*) lungo il vettore Y_B^L (*leader*)
 - $\Delta z(k)$ funzione posizione, relativa dell'origine della terna assi corpo F_B^W (*wingman*) lungo il vettore Z_B^L (*leader*)

- $\Delta x(0)$ componente x del vettore posizione, relativa dell'origine della terna F_B^W rispetto alla terna F_B^L
- $\Delta y(0)$ componente y del vettore posizione, relativa dell'origine della terna F_B^W rispetto alla terna F_B^L
- $\Delta z(0)$ componente z del vettore posizione, relativa dell'origine della terna F_B^W rispetto alla terna F_B^L
- Vettore velocità di volo (*orientato secondo l'asse X_B della terna assicorpo*)  non modificabile a causa della sincronizzazione implicita dell'inseguimento di traiettoria
- Vettore angolo di rollio R

L'inseguimento di traiettoria è possibile solo se del velivolo che si vuole inseguire sia già stata definita la traiettoria.

È necessario scegliere il velivolo da inseguire, definire i parametri di inseguimento ed individuare il tratto da inseguire selezionandolo direttamente sulla traiettoria del leader.

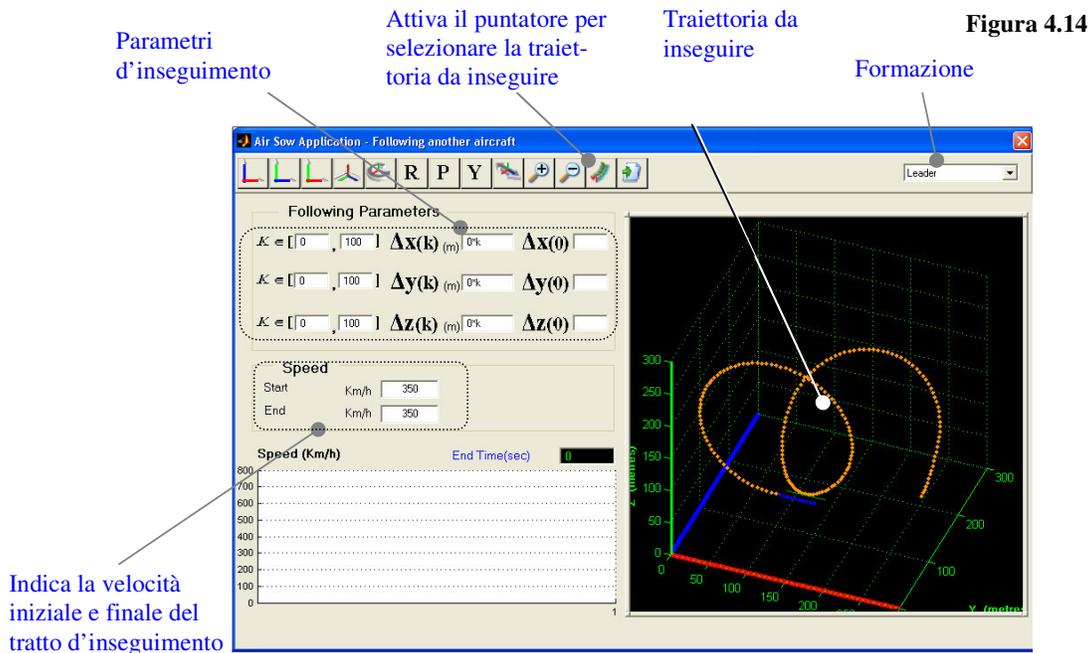
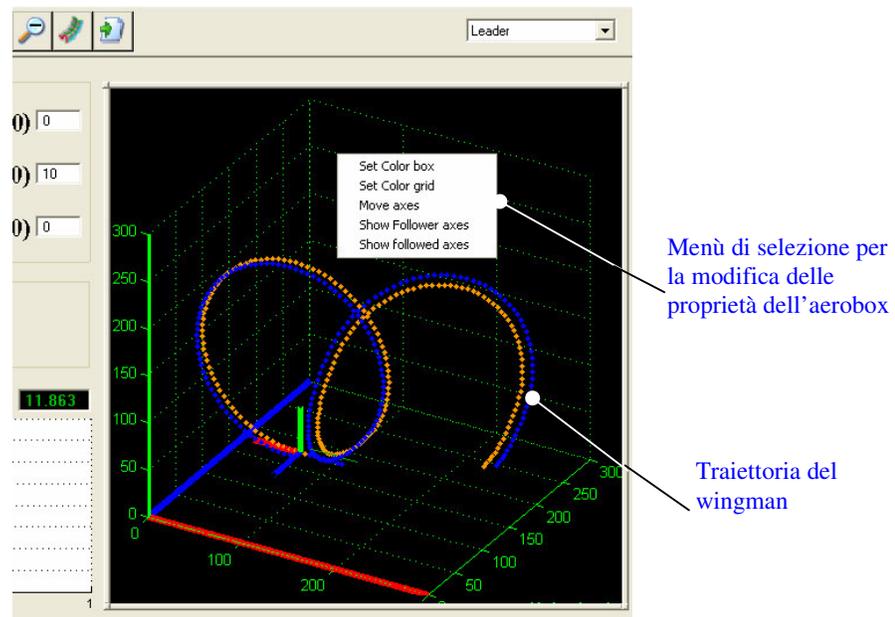


Figura 4.14

Figura 4.15



I parametri di inseguimento permettono di specificare la posizione dell'origine della terna assi corpo del wingman rispetto alla terna assi corpo del leader . Nel caso si inseriscano delle funzioni $\Delta x(k)$ $\Delta y(k)$ $\Delta z(k)$ è necessario definire l'intervallo $[a , b]$ su cui valutare le funzioni.

Nel caso la traiettoria del *wingman* sia già stata in parte definita, la traiettoria di inseguimento proseguirà dal punto di fine della precedente, (*vincolo di continuità*).

Nel punto di attacco della traiettoria d'inseguimento è assicurata la continuità della posizione dell'origine della terna assi corpo, per l'orientazione (*angolo di pitch e angolo di yaw*) sarà l'utente a fare in modo, in accordo con le caratteristiche del velivolo, che non ci siano variazioni discontinue, definendo opportunamente le funzioni di inseguimento.

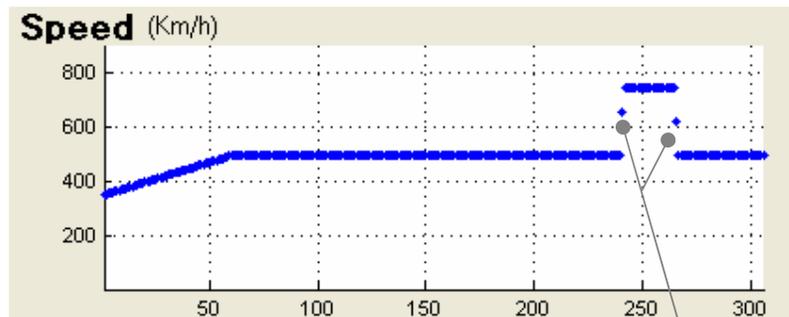
Durante la fase implementativa della procedura di calcolo della traiettoria di inseguimento si è tenuto conto dei seguenti vincoli:

- La durata della manovra di inseguimento deve coincidere con quella della manovra del *leader* adeguando opportunamente il vettore velocità, non modificabile dall'utente.

L'adeguamento del modulo della velocità di volo, poiché calcolato punto per punto porta a valori di accelerazione molto elevati al di fuori delle possibilità del velivolo.

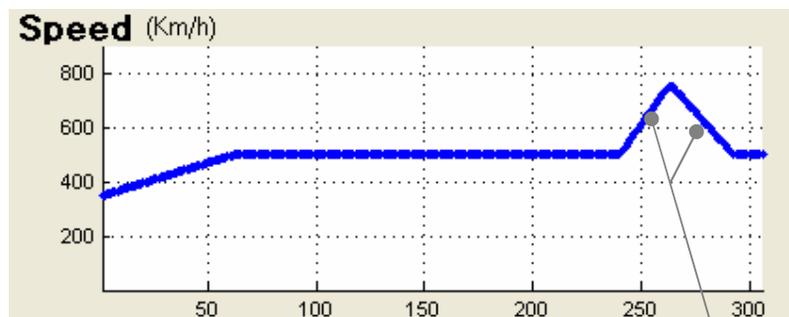
Tali incongruenze sono state eliminate grazie alla procedura di limitazione dell'accelerazione del velivolo (*come nel caso della procedura assistita di setting della velocità degli altri elementi base*). Nella figura 4.16 e 4.17 è possibile notare la differenza tra il modulo della velocità di volo calcolata senza imporre le opportune limitazioni e il modulo della velocità di volo calcolata imponendo i vincoli.

Figura 4.16



Valori non ammissibili di accelerazione

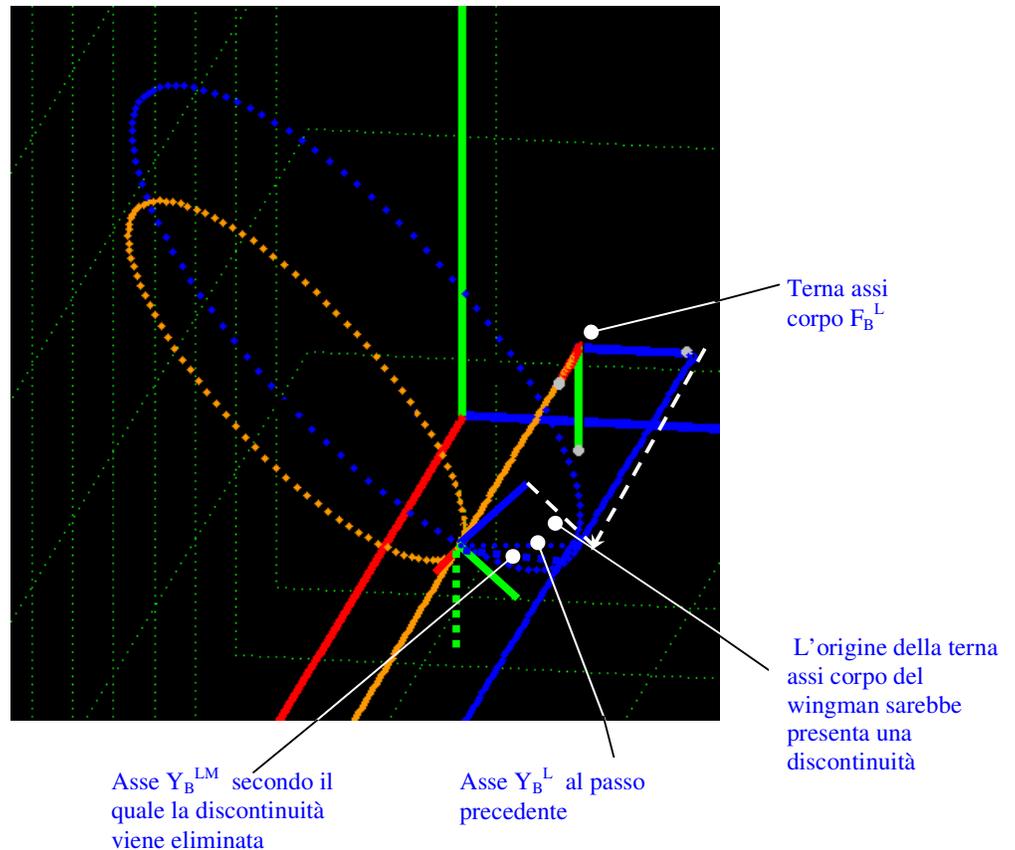
Figura 4.17



Valori ammissibili di accelerazione imposti dalla procedura di limitazione

- Evitare discontinuità della traiettoria calcolata, dovute a particolari configurazioni della traiettoria da inseguire. (fig. 4.18)

Figura 4.18



Nella figura 4.18 la traiettoria del leader è composta da un loop inclinato di un angolo Φ rispetto ad un loop giacente nel piano X_B-Z_B della terna assi corpo del velivolo, valutata alla fine del tratto precedente.

Se i punti della traiettoria del *wingman* fossero calcolati semplicemente valutando l'andamento delle funzioni di inseguimento nella terna assi corpo del *leader*, la procedura produrrebbe una traiettoria discontinua (*discontinuità dell'origine della terna assi corpo del wingman vedi fig. 4.18*).

I punti della traiettoria di inseguimento nel caso riportato in figura 4.18 vengono calcolati valutando le funzioni di inseguimento non nella terna assi corpo del *leader* ma in una terna F_B^{LM} il cui asse Y_B^{LM} giace nel piano individuato dall' asse X_B^L corrente e dall' asse Y_B^L del passo precedente.

4.6 Modificare gli elementi base della sequenza

Aspetto molto importante è costituito dalla possibilità di modificare gli elementi base della sequenza in qualsiasi posizione essi si trovino.

La modifica di un elemento base consiste:

- Nel richiamare il modulo di definizione del blocco elementare con i parametri del blocco da modificare.
- Nella definizione dei parametri per la modifica (*vedi aggiunta di un blocco base*).
- Nell'aggiornamento della sequenza di blocchi successivi al blocco modificato.
 - Vincoli di continuità
 - Vincoli sul vettore velocità di volo

La modifica è permessa solo per

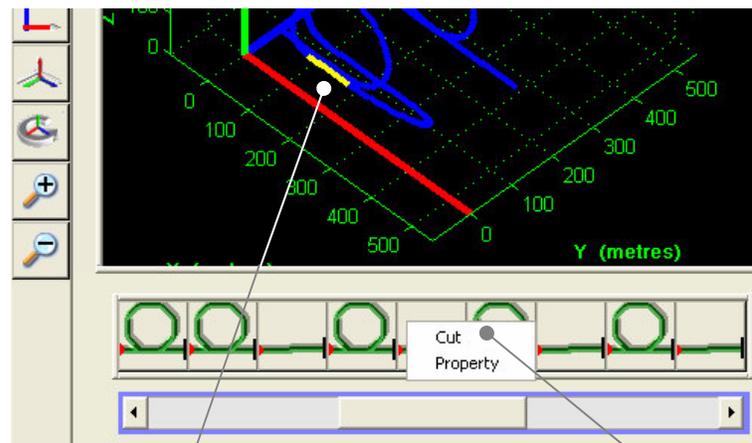
- Line block
- Circle block

Per i *follow block* è prevista la funzionalità di modifica solo nel caso sia l'ultimo blocco della sequenza.

Modificare una sequenza di blocchi elementari significa anche eliminare un blocco ed ovviamente ciò comporta l'aggiornamento della lista di handle grafici e della sequenza di blocchi, successiva al blocco eliminato.

Per poter permettere all'utente di avere a disposizione l'intera sequenza anche in presenza di un numero elevato di blocchi elementari ho realizzato l'oggetto *strip control*.

Figura 4.18



Cliccando sull'handle grafico corrispondente all'elemento base viene evidenziato il tratto corrispondente della traiettoria

Uicontextmenù associato ad ogni handle grafico che descrive la sequenza.

Le funzioni associate all'uicontextmenù permettono di eliminare il blocco (*cut*), di visualizzare le proprietà del elemento base e di modificarle (*property*).

Il codice* che adegua la sequenza alla modifica, definisce le proprietà elemento per elemento, utilizzando le proprietà intrinseche del blocco e i vincoli ormai noti.

for index=i+1:m.Numeroblocchi

```

if strcmp(m.Listaproprieta(index).Tipo,'Line');
    [proprieta,continuitap]=setpropertiesline(hObject, eventdata,...
        handles,m.Listaproprieta(index),...
        m.Listaproprieta(index-1),m.continuita(index-1));
    m.Listaproprieta(index)=proprieta;
    m.continuita(index)=continuitap;
end
    
```

Ridefinisce un line block

end

Ridefinisce un
circle block

```

if strcmp(m.Listaproprieta(index).Tipo,'Circle');

    [proprieta,continuitap]=setpropertiescircle(hObject, eventdata,...
        handles,m.Listaproprieta(index),...
        m.Listaproprieta(index-1),m.continuita(index-1));
    m.Listaproprieta(index)=proprieta;
    m.continuita(index)=continuitap;

end
    
```

```

if strcmp(m.Listaproprieta(index).Tipo,'Functionfollow');

    [proprieta,continuitap]=setpropertiesfollow(hObject,eventdata,...
        handles,m.Listaproprieta(index),...
        m.Listaproprieta(index-1),m.continuita(index-1));
    m.Listaproprieta(index)=proprieta;
    m.continuita(index)=continuitap;

end
    
```

end

Ridefinisce un
follow block

4.7 Definire una manovra indipendente

Definire una manovra indipendente dalla progettazione di un air show significa:

- Modificare una manovra indipendente già presente nell'archivio (*fig. 4.19*)
- Progettare una nuova manovra indipendente (*fig. 4.20*)

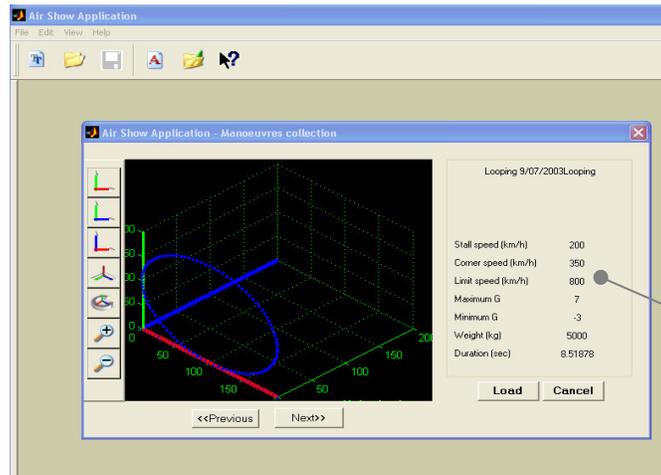
Questa funzionalità permette di risparmiare tempo nella progettazione dei futuri air show, infatti è consigliato all'utente di salvare nell'archivio

delle manovre indipendenti, parti di traiettoria che considera riutilizzabili in futuro.

Progettare una nuova manovra indipendente comporta l'immissione dei parametri di riferimento dell'involuppo di volo.

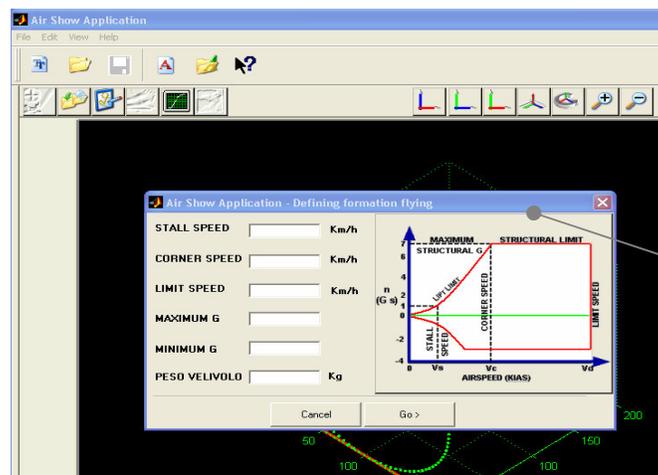
La GUI main per la definizione di una manovra indipendente è leggermente diversa rispetto a quella di progettazione di un air show. Le funzionalità presenti durante la progettazione di un air show rimangono collocate nelle posizioni già note anche se alcune di esse sono disabilitate.

Figura 4.19



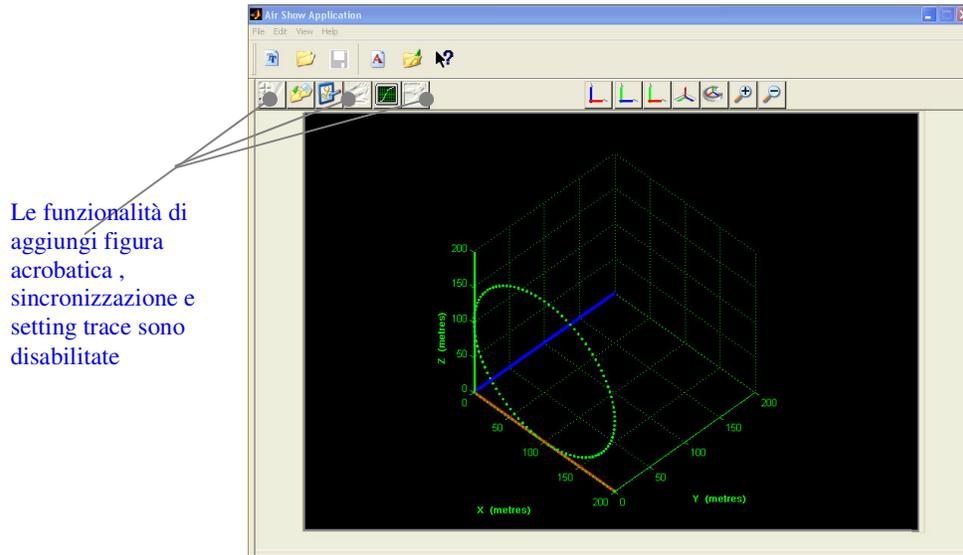
Permette di scorrere l'archivio per selezionare una manovra indipendente già progettata e che si vuole modificare

Figura 4.20



Per definire una nuova manovra indipendente è necessario definire i parametri dell'involuppo di volo di riferimento

Figura 4.21



Progettare una manovra indipendente significa dare la possibilità all'utente di creare degli archivi in cui memorizzare le manovre stesse e da cui caricarle all'occorrenza.

4.7.1 Aggiungere una manovra indipendente ad un archivio di manovre

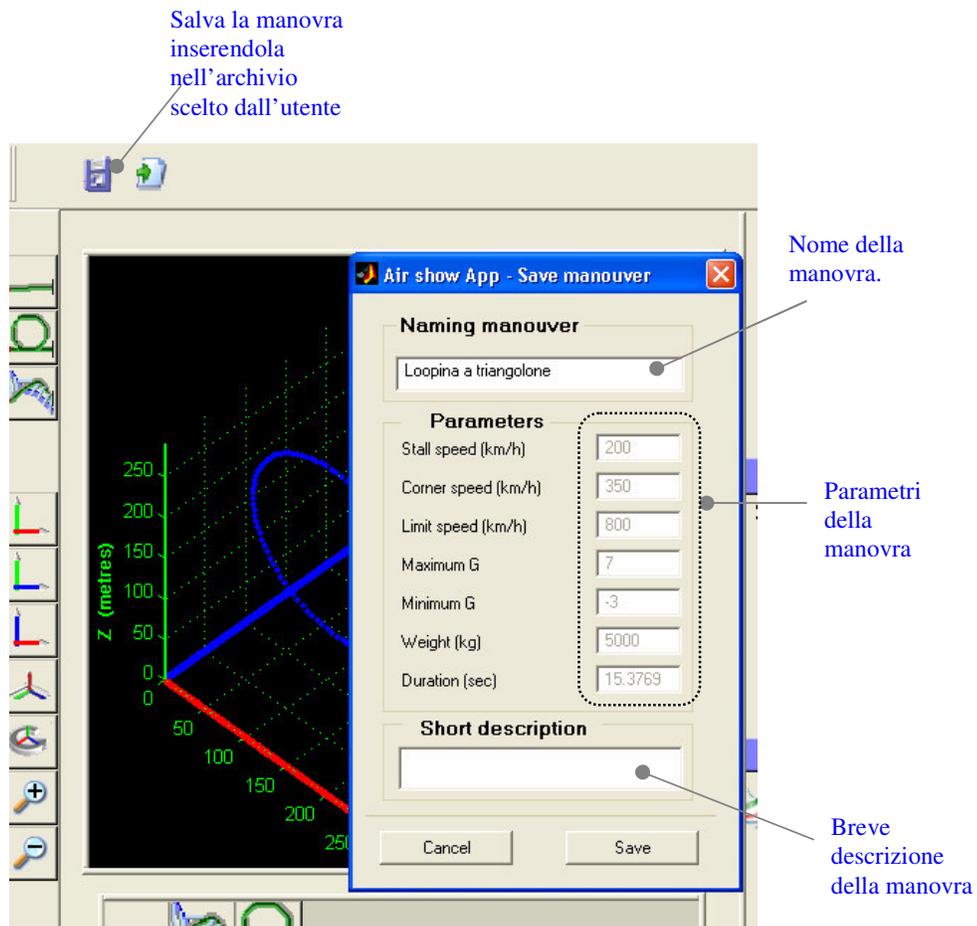
La definizione di una manovra indipendente, ha senso solo dando la possibilità all'utente di creare un archivio in cui poter memorizzare quelle progettate.

L'utente dell'applicazione può all'occorrenza scegliere di salvare la manovra

- In un archivio già esistente
- Creare un archivio nuovo

Oltre a poter definire una manovra indipendente con il modulo descritto precedentemente, è data la possibilità di salvare una qualsiasi manovra riferita ad una figura acrobatica di un velivolo della formazione (*opzione salva del modulo di definizione delle sequenze*) per velocizzare la fase di aggiornamento e/o creazione dell'archivio (fig. 4.22).

Figura 4.22



Scelto l'archivio in cui salvare la manovra, nel caso l'utente immetta un nome di manovra già presente, la procedura avvisa che, proseguendo la manovra verrà aggiornata. La denominazione della manovra costituisce il parametro di identificazione della stessa nell'archivio.

4.8 Aggiungere una manovra indipendente alla sequenza definita

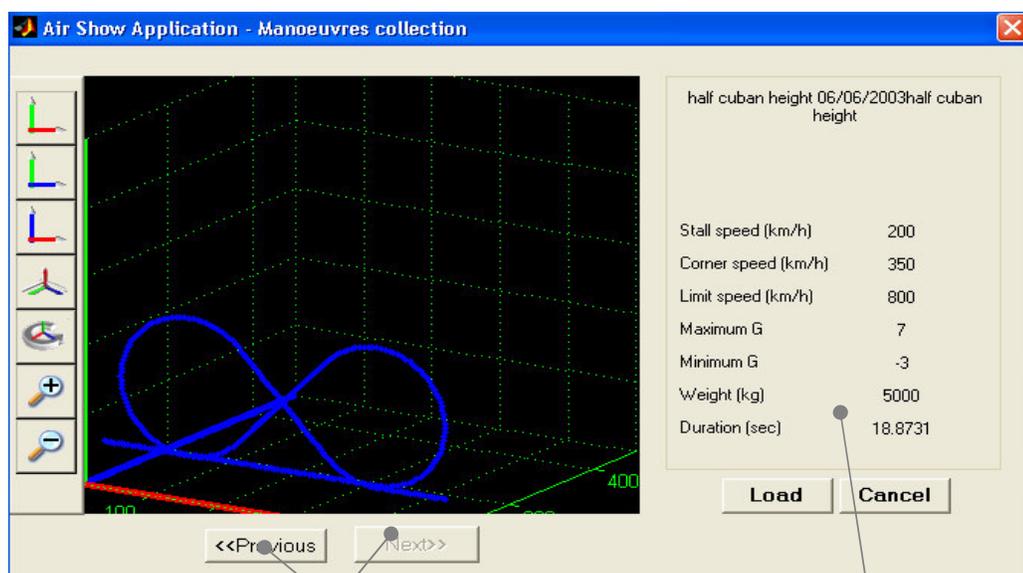
Come detto nel paragrafo precedente l'utente ha la possibilità di definire delle manovre indipendentemente dalla progettazione di un air show. L'utente può aggiungere alla sequenza di definizione una manovra già progettata e salvata in archivio solo se:

- I parametri di definizione della manovra sono identici a quelli impostati per i velivoli della formazione
- I parametri di definizione della manovra sono gli stessi della manovra indipendente che si sta progettando

La manovra viene accodata alla sequenza secondo i seguenti criteri:

- Se la sequenza è vuota la manovra caricata viene posizionata nell'aerobox nella posizione in cui è stata progettata
- Se la sequenza è già in parte definita la manovra viene accodata alla sequenza rispettando i vincoli di continuità

Figura 4.23



Permettono di scorrere sequenzialmente l'archivio

Parametri di manovra

4.9 Definire la durata della scia

La definizione della scia è realizzata da un apposito modulo al quale viene passato come parametro l'intero air show. L'utente deve selezionare un velivolo della formazione e una figura acrobatica.

Le funzionalità previste per la definizione della scia sono:

- Definizione del tratto e del colore
- Elimina scia

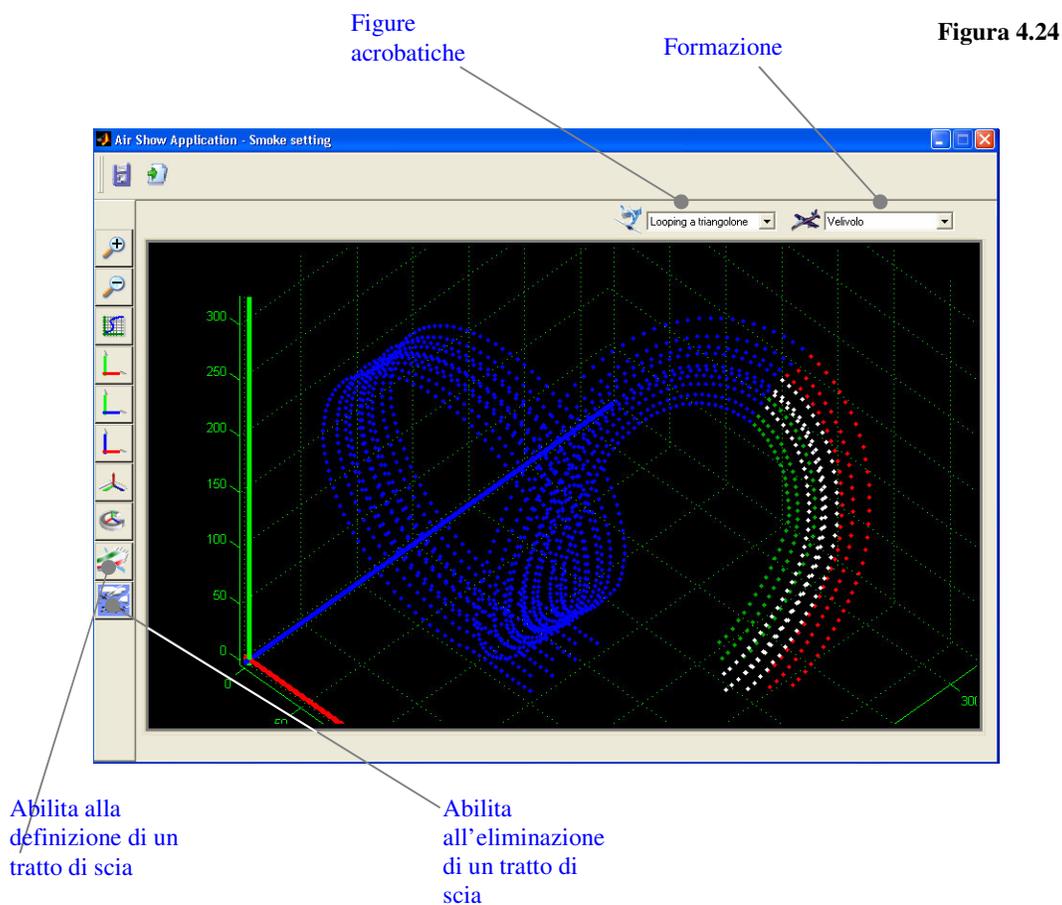


Figura 4.25

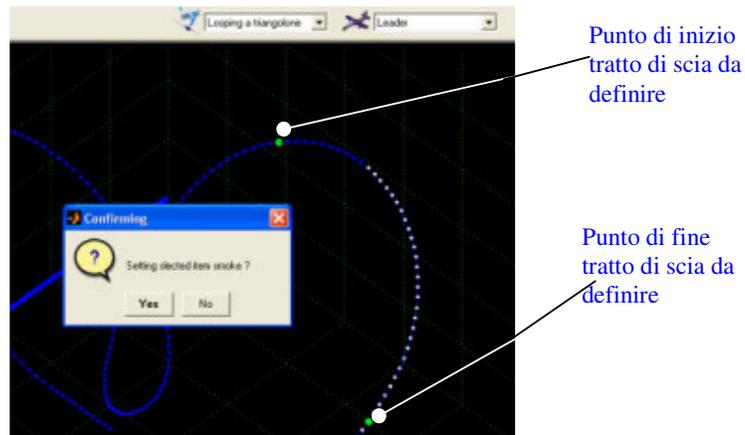
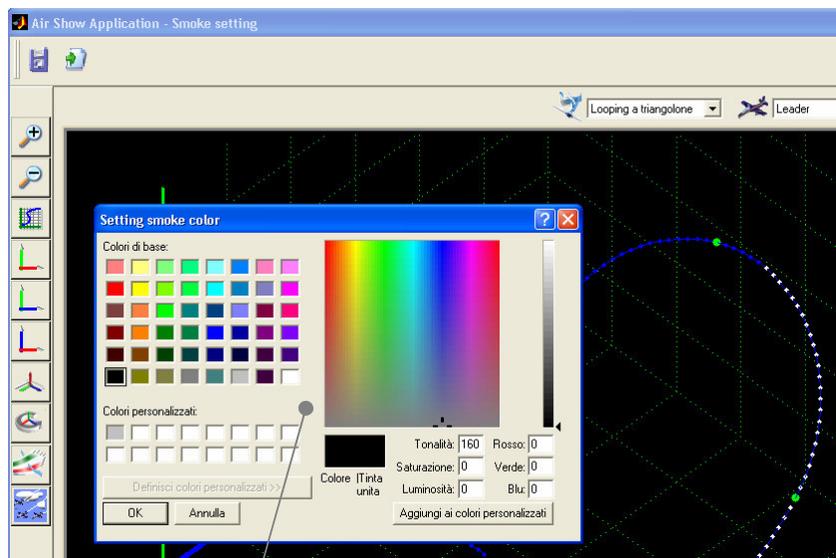


Figura 4.26



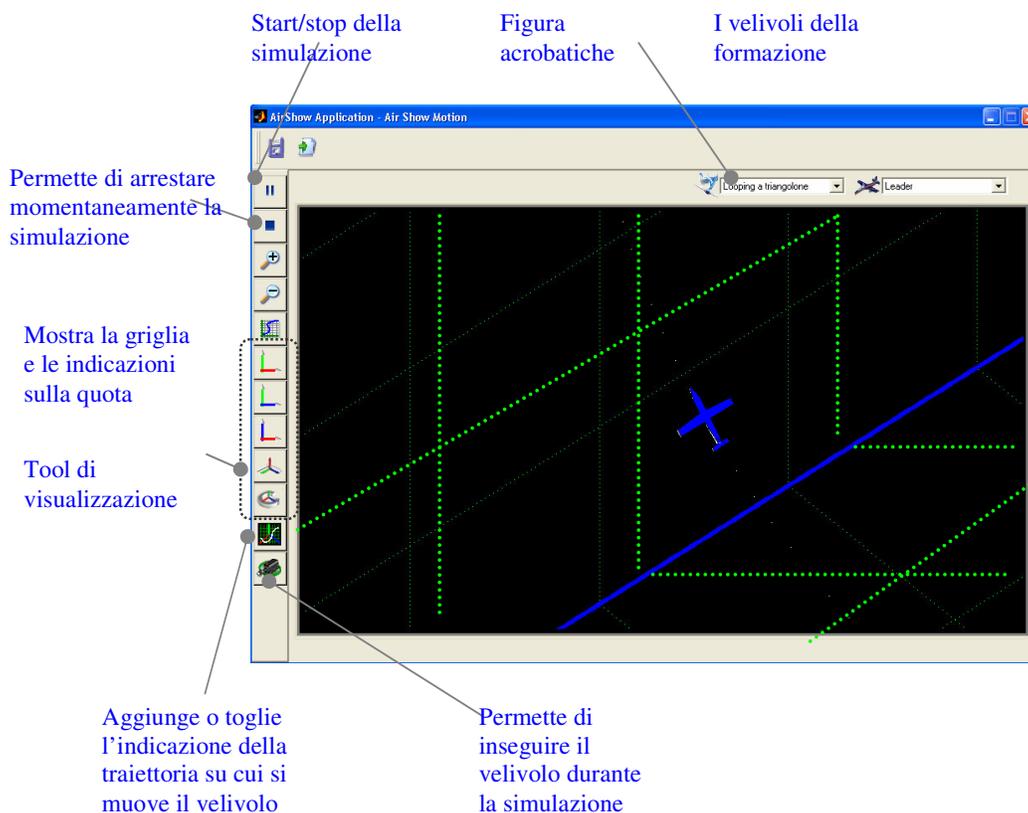
Permette di scegliere il colore della scia

La procedura di eliminazione di un tratto di scia è simile a quella di definizione, ovvero è necessario indicare un tratto di traiettoria selezionando due punti di essa.

4.10 Verificare il comportamento dinamico di un velivolo

Per verifica del comportamento dinamico di un velivolo si intende la valutazione, da parte dell'utente, dell'orientazione e della posizione del singolo velivolo. L'utente grazie a questa funzionalità, (fig. 4.26) può agevolmente verificare la presenza di tratti di traiettoria da modificare secondo le sue aspettative, senza essere vincolato alla progettazione delle traiettorie degli altri velivoli.

Figura 4.27



La funzionalità di verifica del comportamento di un singolo velivolo è utile nel caso si faccia uso frequente del *follow block* che in alcuni casi, a

motivo di errori numerici, potrebbe produrre risultati non perfettamente conformi alle aspettative dell'utente.

Specificare in modo molto preciso una traiettoria, verificando minuziosamente il comportamento del velivolo, risulta utile in fase di definizione della traiettoria d'inseguimento, in quanto, tanto più accurata è la definizione della traiettoria da inseguire (*velocità di volo, angolo di rollio inizio/fine scia*), tanto più precisa sarà quella prodotta dalla procedura di inseguimento.

Per migliorare la visualizzazione del velivolo il modulo permette di inseguire il velivolo, il quale rimane sempre nel centro dell'area di *rendering*.

L'inseguimento del velivolo  può essere attivato in concomitanza delle funzionalità di scelta della vista (*piani di visualizzazione*).

```
if getappdata(handles.Main,'Segui')==1
    set(handles.vista3d,'CameraTarget',[x,y,z]);
end
```

Il codice sopra riportato indica che l'inseguimento è implementato aggiornando la proprietà *'CameraTarget'* dell'aerobox, con la posizione del baricentro del velivolo. Tuttavia nel caso la funzione di zoom non riesca ad evidenziare i particolari della simulazione, selezionando il velivolo viene attivato l'inseguimento dell'oggetto (*non della posizione del baricentro*).

```
aereo=patch('Vertices',vertex_matrix,'Faces',faces_matrix,...
    'FaceColor','w');
set(aereo,'ButtonDownFcn','smokeform("zooaereo_callback",gcbo,[],...
    guidata(gcbo))');
```

Alla proprietà *'ButtonDownFcn'* dell'oggetto *aereo** è associata la funzione *zooaereo_callback(gcbo, [],guidata(gcbo))* che chiama la routine MATLAB *camlookat(hObject)*.