



Approcci multiagente al controllo del traffico aereo

Guagneli Luigi

luigi.guagneli@studio.unibo.it

II Facoltà di Ingegneria

Cesena

OUTLINE

- Introduzione
- Contestualizzazione del problema
- Primo approccio
- Secondo approccio
- AgentFly
- Conclusioni

Introduzione:

Definizione e valutazioni sull'introduzione della computerizzazione nel controllo del traffico aereo

Air Traffic Control (ATC)

- Servizio fornito da controllori a terra che dirigono gli aerei sulla terra e in aria.
- Scopo principale: separare gli aeromobili per evitare collisioni, per organizzare e accelerare il flusso del traffico, e per fornire informazioni e supporto per i piloti quando possibile
- prevenzione delle collisioni: ***separation***
- “spazio aereo controllato”

A seconda del tipo di volo e la classe di spazio aereo, l'ATC può impartire

- ***Instructions***
- ***flight information*** (advisories)

Computerizzazione dell'ATC

- “Sistema di trattazione dell'informazione in tempo reale” – sistema “fonetico-manuale” puro in cui le operazioni di controllo per lo più di natura intellettuale, sono effettuate in maniera rigidamente coordinata e in tempo reale da un gruppo di individui operanti in “equipaggi”
- Limiti di capacità del sistema → suddivisione dei compiti assegnati
- Attualmente i Sistemi usati nei centri ATC sono il “**Procedurale-Radar**”
- Computerizzazione:
 - a) riduzione dei carichi di lavoro mediante l'esecuzione di operazioni attualmente eseguite dai controllori o dai loro assistenti;
 - b) eliminazione degli errori umani;
 - c) esecuzione di operazioni che superino le capacità dei controllori, sottoposti alla pressione di un'alta densità di traffico.

Contestualizzazione del problema

Incremento del traffico aereo e Approcci tradizionali

- Decisioni di routing e condizioni aeroportuali → 1.682.700 ore di ritardo nel 2007
- Gestione del traffico aereo attuale → sistema centralizzato, strategia di routing gerarchico in grado di eseguire proiezioni di flussi che vanno da 1 a 6 ore

Approcci tradizionali:

- **Modello Lagrangiano** calcolo delle traiettorie dei singoli aeromobili (evitare le collisioni, ridurre la congestione)
- **Modelli Euleriani** pattern di flusso nello spazio aereo

Difficile applicazione a situazioni in cui le condizioni cambiano localmente o l'automazione lavora ancora con i controllori umani

Algoritmi centralizzati
Soluzione monolitica

- lenti
- difficilmente distribuibile
- difficoltà di esecuzione del “**sanity check**”

Controllo del traffico Aereo basato sugli agenti

- Agenti autonomi cercano di ottimizzare alcuni goal generali attraverso l'apprendimento o la negoziazione
- Interazioni ispirate da principi economici → equità nella gestione del traffico aereo attraverso un sistema artificiale monetario che permette ritorsione contro gli agenti avidi
- **Learning Agent** con calcolo di ricompense che incoraggino la cooperazione
- **AgentFly** → controllo del flusso del traffico aereo affidabile senza pianificazione centralizzata

Primo approccio

Configurazione dello spazio aereo

Flusso regionale e nazionale → da 20 min – 8 ore

Facet (Future ATM Concepts Evaluation Tool): modello basato sulla fisica dello spazio aereo USA. Strumento affidabile per la simulazione del traffico aereo, ampiamente usato dalla FAA, dalla NASA e dall'industria

USA - 20 centri regionali e 830 settori

Due i problemi:

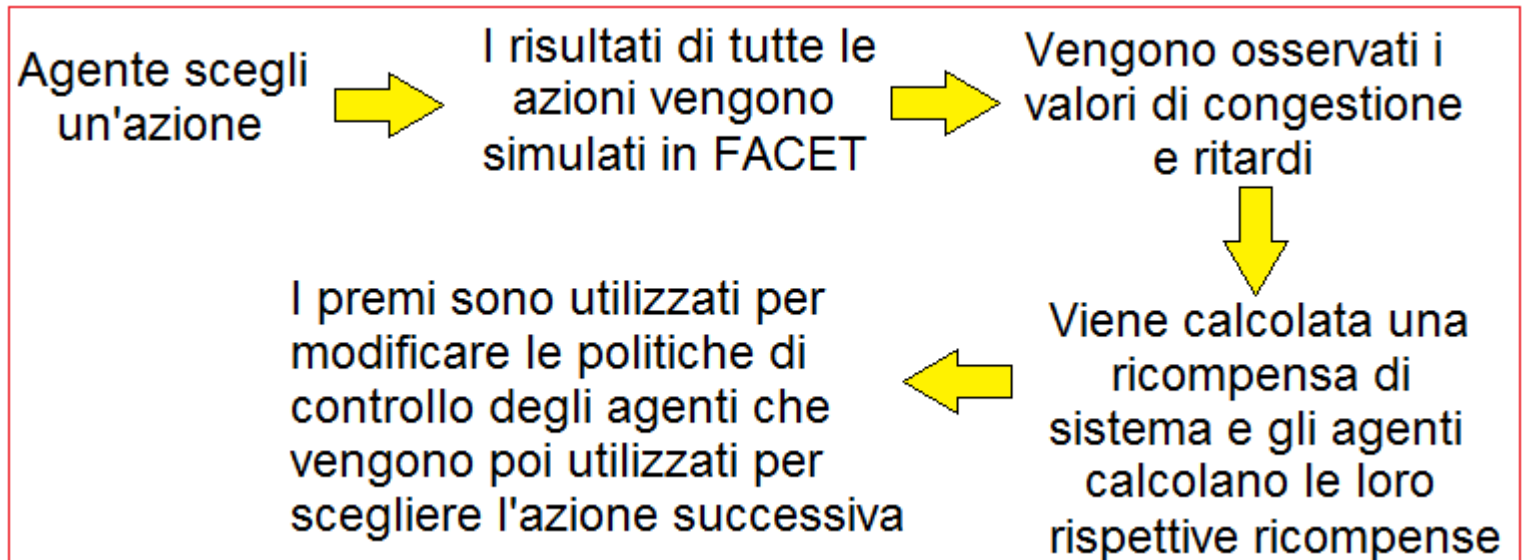
- (i) modellazione e simulazione di un grande sistema complesso (fedeltà e affidabilità dei risultati)
- (ii) stabilizzazione del metodo con cui viene valutata la gestione del flusso (la riduzione diretta del ritardo totale può portare ad *ingiustizie* verso particolari regioni o enti commerciali)

Evoluzione del sistema

Funzione di valutazione

$$G(z) = -(B(z) + \alpha C(z))$$

- $B(z)$ → penalità di ritardo totale per tutti gli aerei nel sistema
- $C(z)$ → penalità totale di congestione
- α → costo di congestione



- Quattro decisioni critiche:
- **definire gli agenti,**
 - **definire lo spazio di azione degli agenti,**
 - **selezionare gli algoritmi di apprendimento,**
 - **selezionare la struttura di ricompensa.**

Definizione degli agenti...

Gli aerei:

- **PRO:**
 - azioni intuitive (cambio piano di volo, incremento o decremento velocità e altitudine)
 - alto livello di granularità (ogni agente la sua politica)
- **CONTRO:**
 - 40.000 aerei = spazio multiagente massiccio
 - campionamento dello spazio di stato lento

I fix (posizioni individuali a terra per tutto lo spazio aereo):

- il numero può variare a seconda delle necessità



- fix stazionari → agenti possono raccogliere dati e riscontrano il comportamento da premiare

- piani di volo di un aereo sono composti da fix → gli agenti possono influenzare i pattern di flusso di traffico.

Ogni agente è responsabile di tutti gli aerei che passano attraverso il suo fix

- fix impiegati all'interno delle attuali procedure di routing di traffico aereo

...e delle azioni

Tre tipi di azione

- ✓ **Miles In Trail (MIT)** - la distanza che il velivolo deve tenere da ogni altro mentre si avvicina al fix. MIT più alto, meno aerei saranno in grado di passare attraverso quel fix durante i periodi di congestione
- ✓ **Ground Delay (Ritardo a terra)** l'aereo arriverà più tardi al fix. Se alcuni agenti scelgono questa azione e altri no, la congestione decremerà perché verrà sparsa.
- ✓ **Rerouting (riindirizzare)** gli aerei che attraversano il suo fix. Deviando alcuni aeromobili lontano da particolari regioni, questa azione può prevenire i casi specifici di congestione e quindi ridurre la congestione totale.

Algoritmi di apprendimento degli agenti

Ogni agente avrà la propria funzione ricompensa e avranno lo scopo di massimizzarla utilizzando un algoritmo di ***reinforcement learning*** (ϵ -greedy learners con tasso di sconto 0).

Ad ogni passo temporale, un agente compie un'azione e riceve una ricompensa valutando tale azione. L'agente userà poi tale ricompensa per aggiornare le sue politiche di azione in modo tale da cercare di intraprendere azioni che portino ad aumentare il premio.

Dopo aver eseguito un'azione e ricevuto una ricompensa R , un agente aggiorna il suo valore per l'azione a , $V(a)$ come segue:

$$V(a) \leftarrow (1-\lambda)V(a) + (\lambda)R$$

dove λ è il tasso di apprendimento. Per promuovere l'esplorazione di nuove azioni, ad ogni passo temporale l'agente sceglie da una tabella l'azione con il valore più alto con probabilità $1 - \epsilon$ e sceglieremo un'azione casuale con probabilità ϵ .

Struttura ricompensa dell'agente

- **Approccio diretto** (ogni agente riceve le prestazioni del sistema come sua ricompensa): non scala nei grandi sistemi
- **Differenza di ricompensa** $D_i \equiv G(z) - G(z - z_i + c_i)$
- **Stima del difference rewards**
$$D_i^{\text{est}} = G_f(f(z)) - G_f(f(z) - E(f(z)|z_i) + E(f(z)))$$
- **Difference rewards precalcolato**
Calcolo a priori della stima della differenza di premio, riducendo così il numero di computazioni

Secondo approccio

Considerazioni sull'approccio ad agenti

Ciascuna **linea aerea** ha **BELIEFS**, **DESIRES** e **INTENTS** relative all'ambiente del traffico aereo, ed una strategia opportuna per raggiungere i propri goal, risulta una buona soluzione trattarli come *Agenti*.

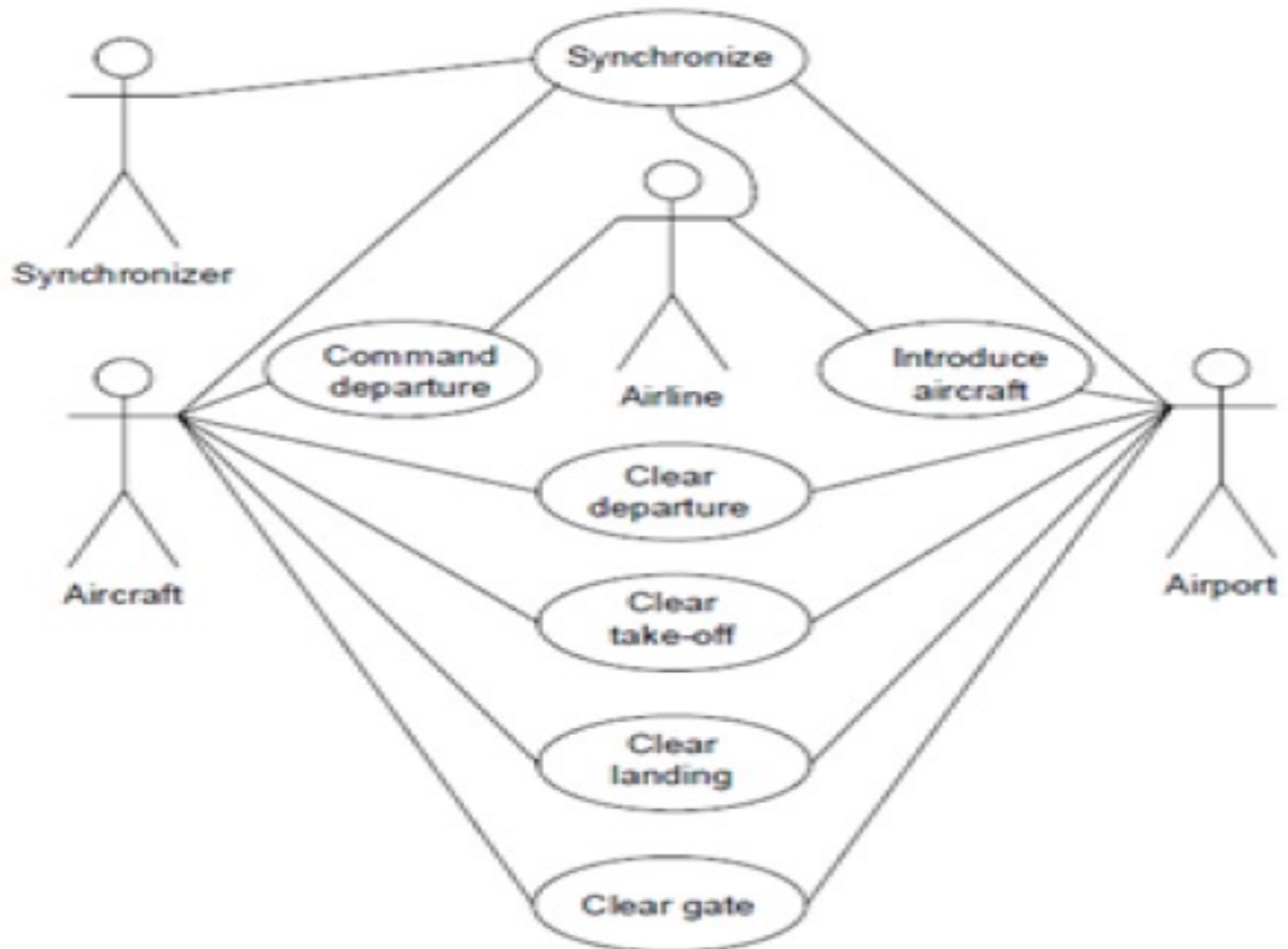
Anche l'**aeromobile**, nonostante sia attivato e regolato da una strategia della linea aerea, ha un alto grado di autonomia quando ci sono da prendere decisioni riguardo la sicurezza del volo.

L'**aeroporto** è visto come un entità autonoma la quale deve controllare l'uso sicuro delle sue risorse, come piste, zone di rullaggio e cancelli d'imbarco.

Tre classi di agenti: **Airline**, **Aircraft** e **Airport**.

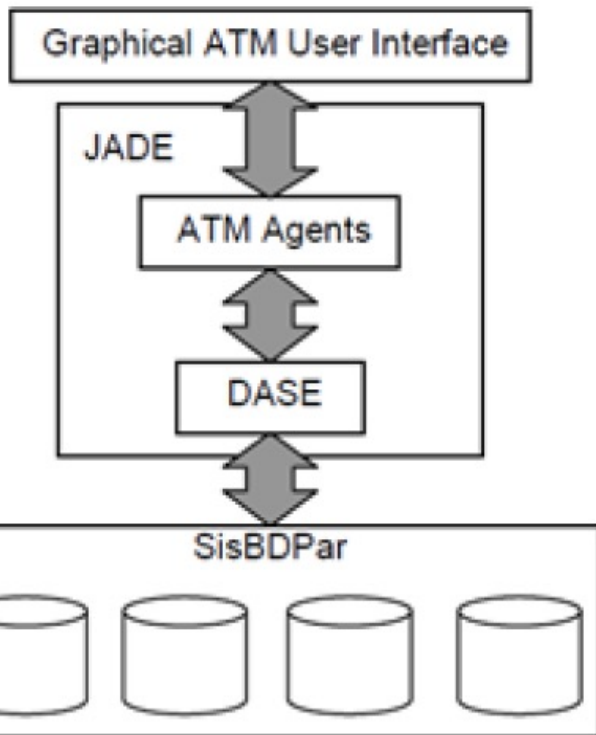
Il passeggero viene visto come un elemento passivo → le decisioni in tempo reale di un volo non sono fatte dai passeggeri.

Definizione



Infrastruttura software

Tre livelli principali:



SisBDPar: un sistema di database parallelo distribuito;
JADE (Java Agent DEvelopment framework): fornisce un'infrastruttura ad agenti;
Graphical ATM User Interface: può essere accesa o spenta, in base agli scopi della simulazione.

All'interno di **JADE**, vengono sviluppati due distinti gruppi di agenti:

Agenti ATM: implementazione del modello astratto di agente

DASE (Distributed data Agent Service Environment): fornisce un grappolo di servizi dati che si integrano armonicamente con la piattaforma multi-agente JADE.

DASE

Obiettivi:

1. Fornire agli agenti un semplice accesso ai dati distribuiti;
2. Offrire risorse aggiuntive correlate all'accesso dati, come la garanzia di consistenza dei dati attraverso il controllo di concorrenza distribuito

La maggior parte dei compiti degli agenti DASE vengono attivati dagli agenti client, al di fuori di DASE, e, rappresentati dagli agenti ATM.

Ci sono tre gruppi principali di agenti funzionali:

DASE Kernel Agents: Gli agenti Kernel sono responsabili dell'avvio, lo spegnimento, il controllo e il monitoraggio del sistema DASE.

DASE Data Access Agents: Questo gruppo ha gli agenti che trattano con i client per le operazioni più frequenti di accesso ai dati (dati ambiente, database).

DASE Concurrency Control Agents: Gli agenti in questo gruppo garantiscono la coerenza dei dati in condizioni di concorrenza.

Terzo approccio: AgentFly

Simulazione, Pianificazione e Collision Avoidance (1/2)

Ogni attività di volo → contenitore software per agenti intelligenti multipli.

Ciascun agente: o modella una **funzionalità hardware *specific*** (capacità sensoriale, controllo di volo dinamico), o di **comunicazione**, o incapsula un'intelligenza in grado di prendere decisioni che supporti pianificazione o CA.

AgentFly ha un'architettura che supporta tre principali casi d'uso:

- modellazione multi agente e simulazione di volo libero
- piattaforme di controllo di volo libero senza equipaggio aereo
- Approcci alternativi alla pianificazione, che supporti il traffico aereo civile

Simulazione, Pianificazione e Collision Avoidance (2/2)

- ✧ Non necessita di componenti centralizzate.
- ✧ È basato sulla tecnologia multi-agente **Aglobe**.
- ✧ Rilassa il problema di pianificazione ed esegue le simulazioni di volo multi agente.

Invece di pianificare un'operazione anticollisione per numerosi aerei, viene costruito un piano di volo per ogni velivolo senza considerare possibili collisioni.

Viene simulata tale operazione nell'ambiente AgentFly, che identifica possibili collisioni, e le risolve o attraverso **ripianificazione individuale** o negoziazione **peer-to-peer**.

Pianificazione

Ogni agente → un pianificatore di percorso che fornisce una traiettoria del piano di volo regolare, che rispetta tutti i vincoli e goal del modello dell'aereo.

Ciascun pianificatore → trasforma i goal in una sequenza di **waypoints**, e prepara descrizioni dettagliate del singolo corridoio di volo.

La traiettoria di volo data può essere eseguita dall'integrazione dello standard **Required Navigation Performance (RNP)** in un processo di pianificazione, specificando in tal modo deviazioni ammissibili delle posizioni orizzontale e verticale dell'aereo all'interno del proprio corridoio.

Quando un livello RNP non è più raggiungibile, il pianificatore cerca un altro piano.

Fasi di pianificazione

Il processo di pianificazione del percorso esegue due fasi accoppiate:

1^ fase: il pianificatore prepara la parte **spaziale** del piano di volo, rispettando le restrizioni di traiettoria utilizzando l'algoritmo Accelerated A* (AA*). Lo spazio aereo, viene poi conservato in una struttura ad albero.

2^ fase: il pianificatore pianifica i cambiamenti di velocità di crociera, mappandoli sulle parti di traiettorie spaziali preparate. Il goal è adattare tutte le restrizioni di tempo e velocità di crociera date. Le fasi spaziale e temporale sono connesse in un loop così che il sistema possa manipolare casi estremi.

Collision Avoidance

Quattro classi di meccanismi di CA:

- **Rule-Based CA (RBCA)**

Algoritmo dominio-dipendente basato sul FAA's Visual Flight Rules (VFRs). L'agente determina il tipo di collisione sulla base dell'angolo tra i vettori direzione degli aerei coinvolti. Ogni tipo di collisione = manovra fissa predefinita

- **Iterative Peer-to-Peer Collision Avoidance (IPPCA)**

Negoziazioni multiagente volte a trovare la manovra CA pareto-ottimale

- **Multi-Party Collision Avoidance (MPCA)**

Estende i primi due algoritmi permettendo diverse attività per negoziare le loro manovre collettive CA.

- **NonCooperative Collision Avoidance (NCCA)**

Supporta CA quando la comunicazione tra attività è impossibile

Conclusioni

La gestione efficiente, sicura ed affidabile del flusso del traffico aereo è un problema complesso, che richiede soluzioni che integrano le politiche di controllo con orizzonti temporali che vanno da pochi minuti no ad un anno.

Nell'analisi si è visto come lo studio di una soluzione che utilizzi gli agenti, vada ad intaccare ambiti multidisciplinari (ricerca operativa, intelligenza artificiale, semplice applicazione algoritmica)

L'utilizzo di sistemi multi agenti rappresenta sicuramente una buona base per lo sviluppo di algoritmi di facile implementazione e il cui utilizzo si affianchi alle attuali procedure, senza la necessità di modifiche radicali.