

Architettura Software di un Sistema di Controllo per Multirobotore

Corrado Santoro

ARSLAB - Autonomous and Robotic Systems Laboratory

Dipartimento di Matematica e Informatica - Università di Catania, Italy

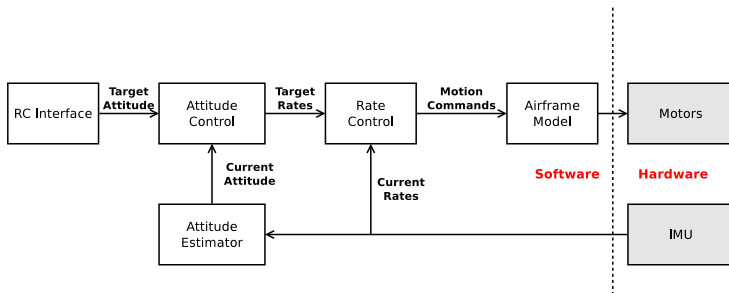
santoro@dmi.unict.it



Programmazione Sistemi Robotici

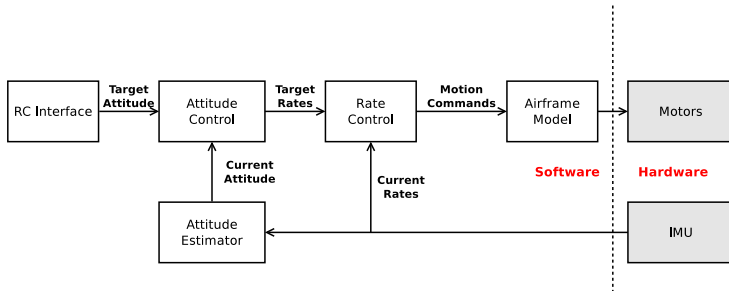
Schema di Controllo

- Lo schema di controllo **in assetto** è rappresentato dalla seguente figura:



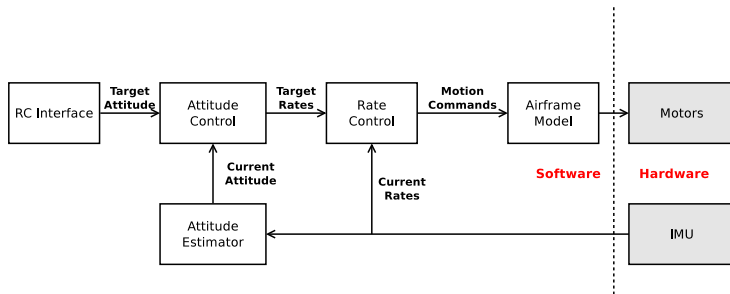
Schema di Controllo

- **RCInterface**, rappresenta l'interfaccia hardware/software con il radio comando; l'output di questo blocco è costituito dai valori di **roll**, **pitch** e **yaw desiderati**, nonché dal valore della **potenza di propulsione (thrust)**



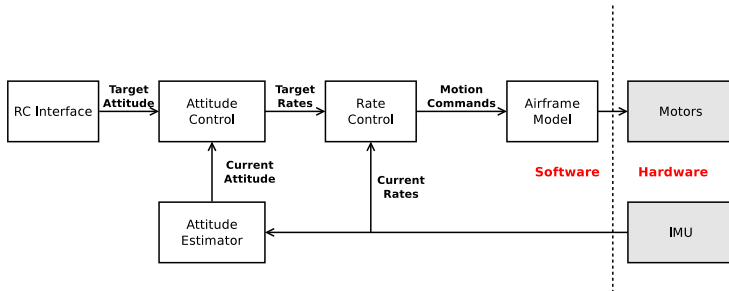
Schema di Controllo

- **AttitudeEstimator**, è l'algoritmo di *sensor fusion* che, a partire dalle informazioni acquisite dalla IMU (Inertial Measurement Unit - giroscopi, accelerometri, magnetometri), stima l'**assetto effettivo** del velivolo



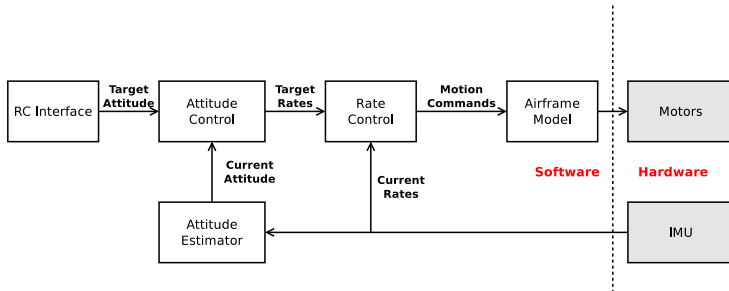
Schema di Controllo

- **AttitudeControl**, implementa i **controllori di assetto** (essenzialmente **proporzionali**), che si basano sulla differenza tra i valori desiderati e stimati degli angoli di Eulero; l'output di ogni controllore è il rispettivo **valore target della velocità angolare**



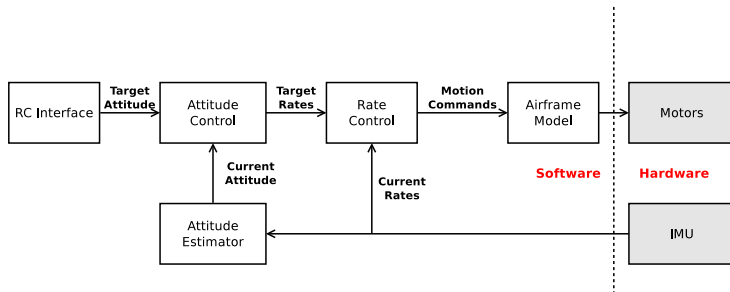
Schema di Controllo

- **RateControl**, implementa i **controllori di velocità** (essenzialmente **proporzionali-integrali**), che si basano sulla differenza tra i valori desiderati ed effettivi delle velocità angolari; l'output di ogni controllore è il rispettivo **comando di rotazione per il blocco motori**



Schema di Controllo

- **Airframe Model**, implementa la traduzione dei comandi di rotazione nei rispettivi comandi per i motori, basandosi sulla configurazione del velivolo (quadrotor-“X”, quadrotor-“+”, hexarotor, octarotor, ...)



Stima dell'assetto

- La **stima dell'assetto** è uno dei maggiori problemi dei sistemi di controllo dei multirotori.
- L'obiettivo è ottenere i valori degli angoli di **roll**, **pitch** e **yaw** a partire dalle misure fornite da
 - Giroscopi (velocità angolari)
 - Accelerometri (accelerazioni lineari)
 - Magnetometri (componenti campo magnetico terrestre)
- A tale scopo si utilizzano opportuni algoritmi di **sensor fusion**
- I più noti algoritmi sono i seguenti:
 - **DCM (Direction Cosine Matrix)**
 - **Filtro complementare**
 - **Filtro di Kalman**

Direction Cosine Matrix

- La DCM è definita come la matrice di rotazione tra “Earth reference” e “Body reference” di un corpo rigido il cui assetto è specificato dagli angoli Eulero θ, ϕ, ψ

Direction Cosine Matrix

$$DCM = \begin{pmatrix} c\theta c\psi & s\phi s\theta c\psi - c\phi s\psi & c\phi s\theta c\psi + s\phi s\psi \\ c\theta s\psi & s\phi s\theta s\psi + c\phi c\psi & c\phi s\theta s\psi - s\phi c\psi \\ -s\theta & s\phi c\theta & c\phi c\theta \end{pmatrix}$$

$$s = \sin, \quad c = \cos$$

- Un vettore $v' = (x', y', z')$ in coordinate **locali** può essere tradotto in coordinate **globali** moltiplicandolo per la DCM: $v = DCM \cdot v'$

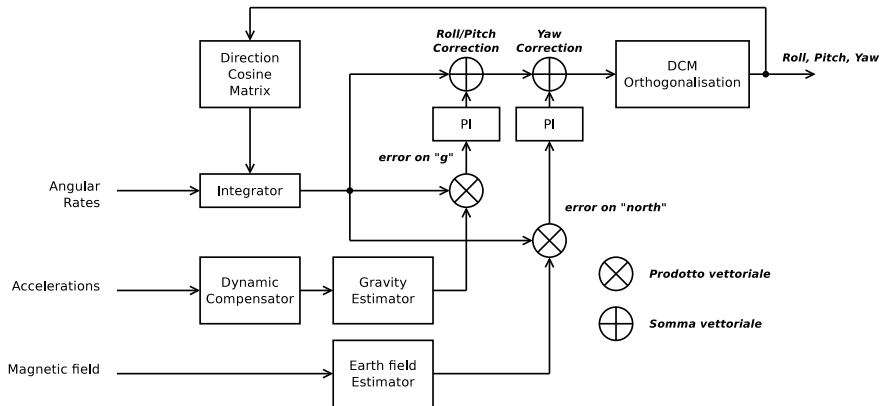
Direction Cosine Matrix

- La DCM ha delle proprietà ben precise
- E' **ortogonale**, la sua trasposta coincide con l'inversa:

$$\begin{aligned}DCM^{-1} &= DCM^T \\ DCM \cdot DCM^T &= I\end{aligned}$$

- L'ortogonalità implica che le colonne (o le righe)
 - sono vettori tra loro ortogonali $\rightarrow$ il **prodotto vettoriale è nullo**
 - sono vettori il cui modulo è **unitario**
- Tali proprietà vanno **sempre** mantenute

L'algoritmo della DCM



Rappresentazione delle rotazioni in 3D

- La DCM è uno dei possibili modi per rappresentare le rotazioni in uno spazio 3D
- Un altro modo è basato sull'uso dei **quaternioni**

Quaternions

Un **quaternione** è un numero complesso con una parte reale e tre parti immaginarie:

$$q = q_0 + q_1\mathbf{i} + q_2\mathbf{j} + q_3\mathbf{k}$$

$$\mathbf{i}, \mathbf{j}, \mathbf{k} = \textit{imaginary units}$$

$$\mathbf{i}^2 = \mathbf{j}^2 = \mathbf{k}^2 = \mathbf{ijk} = -1$$

- Se i numeri complessi permettono di rappresentare una rotazione in 3D, i **quaternioni** permettono di rappresentare le rotazioni in 3D

Rappresentazione delle rotazioni in 3D

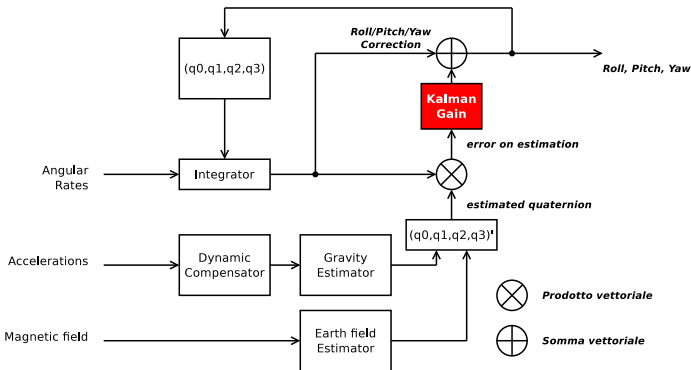
- La rotazione di un vettore $\mathbf{v}$ con i quaternioni si ottiene dal prodotto $\mathbf{q}\overline{\mathbf{v}}\mathbf{q}^*$ dove $\mathbf{q}^*$ è il coniugato di $\mathbf{q}$ e $\overline{\mathbf{v}} = \{0, v_x, v_y, v_z\}$.
- Esistono opportune trasformazioni che permettono di passare dal dominio degli **angoli di Eulero** al dominio dei **quaternioni** (e viceversa):

$$\mathbf{q} \rightarrow (\phi, \theta, \psi)$$

$$(\phi, \theta, \psi) \rightarrow \mathbf{q}$$

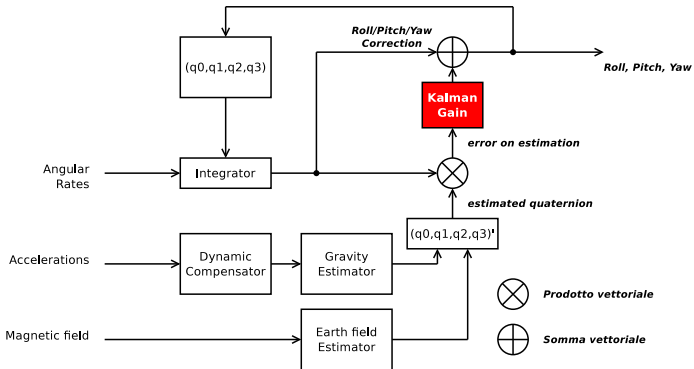
- Gestire le rotazioni, nonché le operazioni di integrazione o derivazione, nell'algebra dei quaternioni è molto più semplice rispetto alla DCM (che invece richiede continue operazioni trigonometriche)
- I quaternioni permettono di risolvere i casi in cui la DCM diventa singolare (e quindi non invertibile)
- Essi sono utilizzati in altri algoritmi di sensor fusion come il **Kalman Filter**

Il filtro di Kalman



- Il blocco **Kalman Gain** non è altro che un “semplice” controllore proporzionale il cui guadagno viene però **ricalcolato ad ogni iterazione**, sulla base di un algoritmo che (note le covarianze degli errori di misura) permette di minimizzare l’errore finale di misura

Il filtro di Kalman



- L'algoritmo si basa su calcoli matriciali e inversioni di matrici, pertanto può risultare particolarmente “pesante”
- Il Filtro di Kalman è tuttavia il miglior algoritmo di sensor fusion a oggi conosciuto ed è applicato in svariati contesti

Architettura Software di un Sistema di Controllo per Multirobotore

Corrado Santoro

ARSLAB - Autonomous and Robotic Systems Laboratory

Dipartimento di Matematica e Informatica - Università di Catania, Italy

santoro@dmi.unict.it



Programmazione Sistemi Robotici