Rotor Model Identification System

Sinossi

Questo progetto rappresenta un tool per l'esecuzione di routines su ESC (Electronic Speed Controller) e la raccolta e l'analisi dei dati della telemetria raccolti durante l'esecuzione.
Grazie alle astrazioni introdotte è possibile utilizzare un modello qualsiasi di ESC. Le istruzioni sono astratte e standardizzate indipendentemente dall'ESC, per questo motivo inoltre routines sono indipendenti dai particolari comandi utilizzati dall'ESC. Con lo stesso principio anche la raccolta dei dati tramite telemetria è indipendente dal modello utilizzato. Su questi dati è possibile definire analisi personalizzate in base alle proprie necessità.

Motivazione

Il progetto è nato dalla necessità di eseguire una serie di accelerazioni a corrente costante su un motore TMotor mt 2212 comandato da un Autoquad ESC32 v2 collegato al pc tramite seriale. Raccolti quindi i dati sulla velocità angolare, la corrente e la tensione del motore, questi vengono utilizzati per il calcolo di parametri propri del motore quali la *torque constant*, la*motor armature constant* e la *internal resistance* sulla base degli studi di Bangura et al. [1].
Con l'obiettivo di permettere il tuning di motori ed ESC diversi, con comandi diversi e necessità di eseguire routine e analisi diverse, abbiamo deciso di sviluppare un'astrazione dell'intero processo, rendendolo così indipendente dal nostro specifico caso e riutilizzabile in un contesto di utilizzo più ampio.

Funzionamento ed utilizzo

Dalla schermata principale è possibile utilizzare i seguenti componenti:

1. *Esecuzione di una routine*

1.a Selezione della porta seriale a cui è collegato l'ESC

1.b Selezione del modello di ESC da utilizzare

1.c Selezione della routine da eseguire

1.d Avvio della routine

2. *Analisi dei dati*

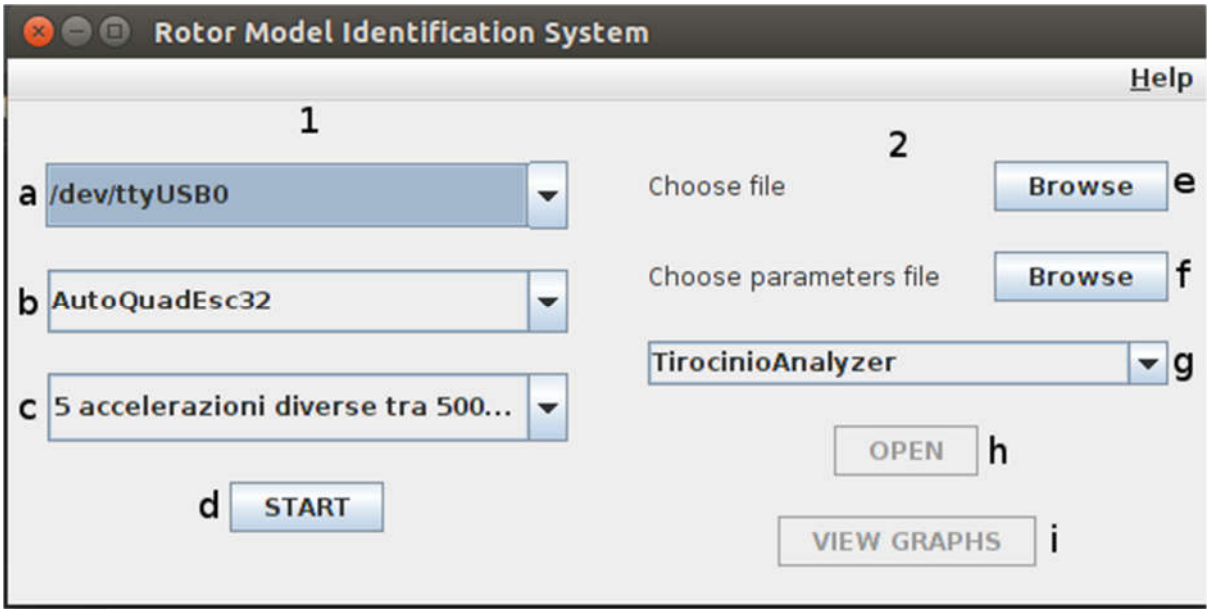
2.f Selezione del file dei dati

2.g Selezione del file di parametri per l'analisi

2.h Selezione del tipo di analisi

2.i Avvio dell'analisi

2.j Ricostruzione dei grafici



Esecuzione di una routine

Selezione della porta seriale a cui è collegato l'ESC

La comunicazione con l'ESC avviene tramite seriale. Se più seriali sono connesse al pc è possibile selezionare quella da utilizzare. La connessione avviene sempre a 230400 baud, 8 bit di dati, 1 bit di stop, 0 bit di parità. In future versioni sarà permesso personalizzare queste impostazioni.

Selezione del modello di ESC da utilizzare

Tutti gli ESC utilizzabili tramite questo programma devono estendere la classe astratta AbstractEsc, saranno così visibili e selezionabili in questo menù a tendina. I requisiti che deve soddisfare

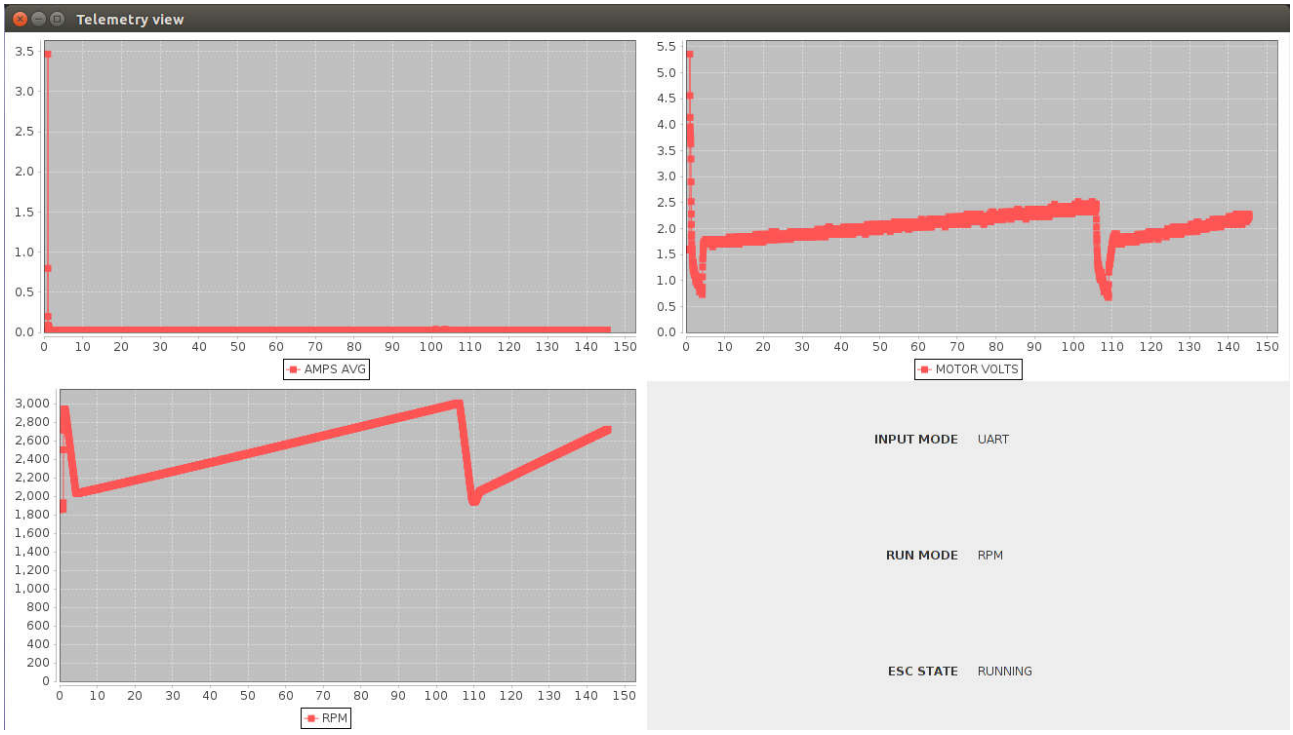
l'implementazione di un ESC sono: gestire l'esecuzione di tutte le istruzioni contenute nella classe Instruction, gestire la telemetria selezionando i parametri da filtrare tra quelli contenuti nella classe TelemetryParameter e inviando i dati attraverso un PipedOutputStream. Come esempio di implementazione di un ESC si veda la classe AutoQuadEsc32.

Selezione della routine da eseguire

Le routines sono contenute in file .rou collocati nella cartella routines nel path in cui è stato eseguito il programma. Tutte le routines ben formate vengono automaticamente parsate all'avvio del programma e presentate in un menù a tendina. Per informazioni su come scrivere una routine si veda più avanti.

Avvio della routine

Se sono stati riempiti correttamente i campi soprastanti questo pulsante permette di avviare la routine sull'ESC. I valori della telemetry specificati nella routine verranno visualizzati in una finestra di questo tipo:



Inoltre vengono salvati in un file chiamato "Motor_data_yyyy-MM-dd_HH-mm-ss.csv" dove i campi del nome rappresentano la data e l'ora dell'esecuzione.

Analisi dei dati

Selezione del file dei dati

Si permette di selezionare un file di dati creato tramite l'esecuzione di una routine precedente. Di questi dati è possibile effettuare delle analisi oppure semplicemte visualizzarne i grafici (solo i valori numerici)

Selezione del file di parametri per l'analisi

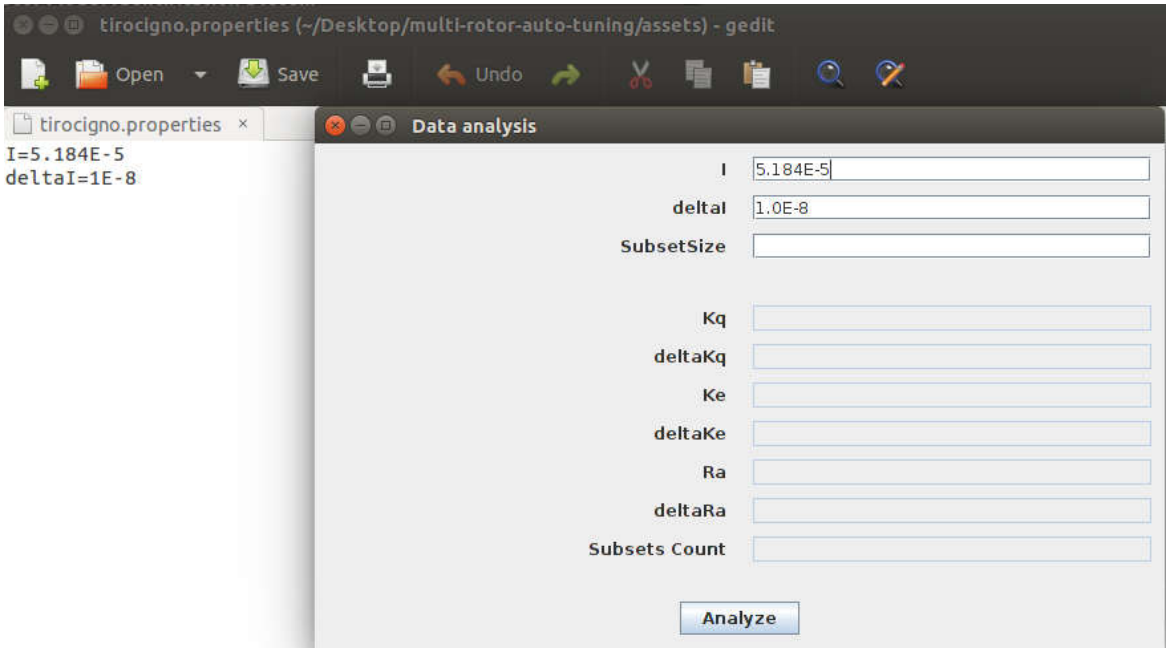
Poichè per l'analisi possono essere necessari ulteriori parametri oltre ai dati della telemetry è possibile, anche se non obbligatorio, indicare qui un file dal quale caricare i valori dei parametri. In questo modo non sarà necessario inserirli a mano ogni volta che si vuole ripetere un'analisi. Il formato dei file segue le convenzioni dei .properties di Java, si veda tirocinio.properties come esempio.

Selezione del tipo di analisi

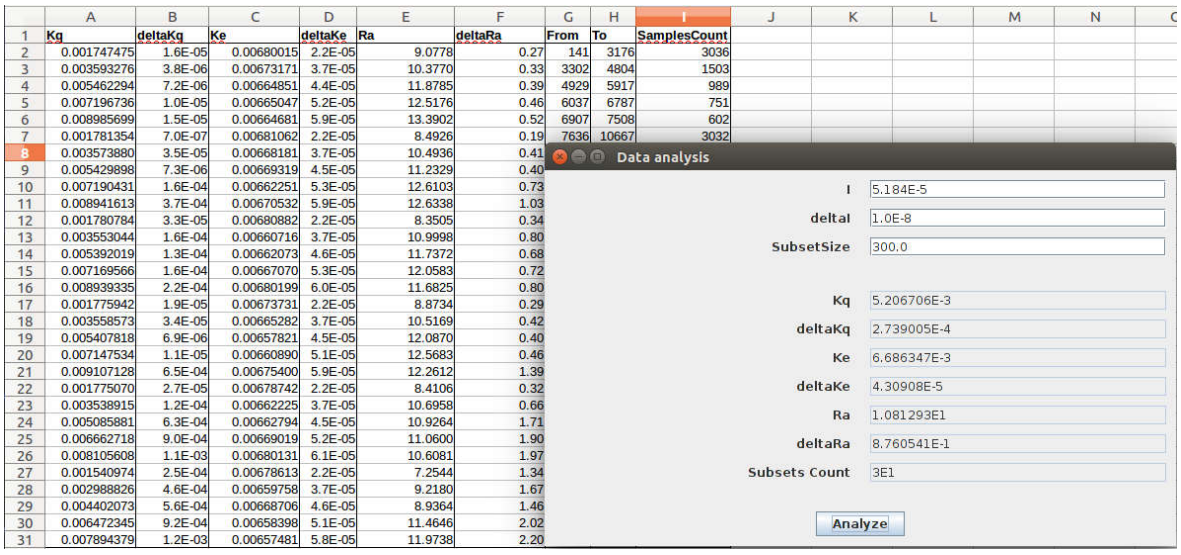
Tutte le analisi si possono utilizzare tramite questo applicativo devono estendere la classe Analyzer, questa fornisce i metodi di base per il caricamento dei dati ed eventualmente dei parametri da file .properties. Le sottoclassi devono implementare solamente il metodo calcola. Un Analyzer, almeno in questa prima versione del programma, è in grado di gestire solamente parametri numerici e fornire in output risultati numerici. Si veda come esempio di implementazione la classe TirocinioAnalyzer (i risultati principali di questa analisi sono mostrati nel pannello grafico, mentre i risultati intermedi utili ad un'analisi più approfondita vengono salvati in un file .csv con lo stesso nome del file di dati seguito dal suffisso "-ANALYSIS".

Avvio dell'analisi

Se sono stati riempiti correttamente i campi soprastanti questo pulsante permette di aprire una finestra di analisi dei dati. In questa finestra sarà possibile modificare i parametri caricati dal file, inserire quelli mancanti e avviare l'analisi. Si noti che questa finestra è generata automaticamente a partire dalla specifica implementazione di Analyzer che si vuole utilizzare, pertanto non è necessario modificare le classi grafiche nel momento in cui si vuole implementare un Analyzer



.



	A	B	C	D	E	F	G	H	I
1	Kq	deltaKq	Ke	deltaKe	Ra	deltaRa	From	To	SamplesCount
2	0.001747475	1.6E-05	0.00680015	2.2E-05	9.0778	0.27	141	3176	3036
3	0.003593276	3.8E-06	0.00673171	3.7E-05	10.3770	0.33	3302	4804	1503
4	0.005462294	7.2E-06	0.00664851	4.4E-05	11.8785	0.39	4929	5917	989
5	0.007196736	1.0E-05	0.00665047	5.2E-05	12.5176	0.46	6037	6787	751
6	0.008985699	1.5E-05	0.00664681	5.9E-05	13.3902	0.52	6907	7508	602
7	0.001781354	7.0E-07	0.00681062	2.2E-05	8.4926	0.19	7636	10667	3032
8	0.003573880	3.5E-05	0.00668181	3.7E-05	10.4936	0.41			
9	0.005429898	7.3E-06	0.00669319	4.5E-05	11.2329	0.40			
10	0.007190431	1.6E-04	0.00662251	5.3E-05	12.6103	0.73			
11	0.008941613	3.7E-04	0.00670532	5.9E-05	12.6338	1.03			
12	0.001780784	3.3E-05	0.00680882	2.2E-05	8.3505	0.34			
13	0.003553044	1.6E-04	0.00660716	3.7E-05	10.9998	0.80			
14	0.005392019	1.3E-04	0.00662073	4.6E-05	11.7372	0.68			
15	0.007169566	1.6E-04	0.00667070	5.3E-05	12.0583	0.72			
16	0.008939335	2.2E-04	0.00680199	6.0E-05	11.6825	0.80			
17	0.001775942	1.9E-05	0.00673731	2.2E-05	8.8734	0.29			
18	0.003558573	3.4E-05	0.00665282	3.7E-05	10.5169	0.42			
19	0.005407818	6.9E-06	0.00657821	4.5E-05	12.0870	0.40			
20	0.007147534	1.1E-05	0.00660890	5.1E-05	12.5683	0.46			
21	0.009107128	6.5E-04	0.00675400	5.9E-05	12.2612	1.39			
22	0.001775070	2.7E-05	0.00678742	2.2E-05	8.4106	0.32			
23	0.003538915	1.2E-04	0.00662225	3.7E-05	10.6958	0.66			
24	0.005085881	6.3E-04	0.00662794	4.5E-05	10.9264	1.71			
25	0.006662718	9.0E-04	0.00669019	5.2E-05	11.0600	1.90			
26	0.008105608	1.1E-03	0.00680131	6.1E-05	10.6081	1.97			
27	0.001540974	2.5E-04	0.00678613	2.2E-05	7.2544	1.34			
28	0.002988826	4.6E-04	0.00659758	3.7E-05	9.2180	1.67			
29	0.004402073	5.6E-04	0.00668706	4.6E-05	8.9364	1.46			
30	0.006472345	9.2E-04	0.00658398	5.1E-05	11.4646	2.02			
31	0.007894379	1.2E-03	0.00657481	5.8E-05	11.9738	2.20			

Ricostruzione dei grafici

Alternativamente alla finestra di analisi dei dati è possibile visualizzare nuovamente i grafici a partire dai dati contenuti nel file. Per ovvi motivi solo i parametri numerici della telemetry vengono riprodotti.

TelemetryParameter

I più comuni parametri della telemetria di un ESC, associati alla stringa che li contraddistingue nell'output di un AutoQuadEsc32 (in altre implementazioni è possibile ignorare questa stringa e mappare la propria telemetria sui parametri astratti nel modo corretto per lo specifico modello)

InstructionType e Instruction

Le più comuni istruzioni che è possibile inviare ad un esc. Alcune di esse probabilmente troveranno un corrispondente diretto nel set di istruzioni del proprio ESC, altre come ACCELERATE andranno gestite come istruzioni di più alto livello e scomposte in comandi più semplici. Alle istruzioni è anche possibile associare ulteriori parametri là dove è necessario, ad esempio per impostare la velocità ad un certo valore.

Routines

Le routines utilizzabili in questo applicativo sono scritte in file con estensioni .rou con il seguente formato:

- la prima riga contiene il nome della routine
- la seconda riga contiene una lista di parametri da utilizzare nella telemetry sottoforma di stringhe separate da virgole(vedere la classe TelemetryParameter per le specifiche stringhe).
- le righe successive devono contenere una valida istruzione per l'ESC, così come specificate nella classe Instruction. Se l'istruzione necessita parametri aggiuntivi questi vanno specificati dopo il nome dell'istruzione e un ':', ogni parametro separato da uno spazio. Le linee vuote e quelle che iniziano per '#' vengono ignorate.

Un esempio di file di routine valido è il seguente:

```
    Rapid acceleration from 2000 to 6000 rpm
    RPM, AMPS AVG, MOTOR VOLTS

    arm
    start
    rpm: 2000
    sleep: 10000
    telemetry: 50
    sleep: 1000

    # accelerate at 1200 rpm/s
    accelerate: 2000 6000 1200
    accelerate: 6000 1000 -400

    sleep: 5000
    stop telemetry
    stop
    disarm
```

Dependencies

Il codice è scritto utilizzando Java 1.7, accompagnato dalle seguenti librerie:

- RxTx http://rxtx.qbang.org
- JFreeChart www.jfree.org/jfreechart
- Commons Math http://commons.apache.org/proper/commons-math
- Reflections https://code.google.com/archive/p/reflections

[1] M. Bangura, H. Lim, H. Kim, and R. Mahony, "Aerodynamic power control for multirotor aerial vehicles", in *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, May 2014, pp 529-536