

Guia de Referência do Projeto Mini drone

Leonardo Toshinobu Kimura

Rogério Nakamashi

INTRODUÇÃO

Essa documentação tem como objetivo servir de auxílio para os futuros trabalhos com o projeto do mini-drone, bem como documentar as hipóteses testadas até então, para evitar o desperdício de tempo e esforço. Serve como tutorial para facilitar o aprendizado do projeto, bem como truques, macetes, e algumas dicas para o trabalho :)

Iniciando

É presumido que a pessoa tenha um conhecimento razoável de python para poder trabalhar nesse programa. Um notebook também é um recurso desejável.

Sobre o mini-drone, é um drone de tamanho bem reduzido (aproximadamente 18g), e tem uma autonomia de 7 minutos de voo, sem carregar nenhum peso extra. Demora em torno de 20 minutos para completar a sua carga, tem uma programação relativamente fácil de fazer (pelo menos em comparação com um drone convencional), e é relativamente robusto, aguentando alguns pequenos impactos. O melhor local para se aprender sobre o drone é o "getting start", do site oficial deles, BitCraze. Lá está explicando direitinho como montar o mini-drone, onde conseguir os programas, e tudo mais.

Sobre o Aplicativo Crazyflie

É um aplicativo simples, acionado por BlueTooth, disponível na playStore. É muito útil para se testar o drone e verificar que está funcionando propriamente, mas não mais do que isso. É quase impossível de se ter um bom controle de voo com ele, nem mesmo manter o drone estável. Não é possível programar nenhum voo, e tampouco ele permite você alcançar a potência máxima dos motores: eu tentei alçar voo com uma câmera, e ele não conseguiu, enquanto com o cfcliente ele conseguiu. Certifique-se de ter o seu android atualizado também :)

Configurando o ambiente de desenvolvimento

Para poder programar o crazyflie é essencial você utilizar o sistema operacional Linux. Nós utilizamos o ambiente Ubuntu, mas você pode utilizar qualquer um, a princípio. Pode ser feito tanto numa máquina virtual (como o virtualBox), quanto no dualBoot, porém parece que na máquina dualBoot é mais adequado, por haver menor delay na transmissão (seja de vídeo, seja de comandos para o drone). Mas fique a sua escolha. No nosso projeto foi utilizado a versão 18.04 de Ubuntu

Nos tutoriais há a opção de se utilizar a OVA deles, pronto para os desenvolvimentos iniciais. Enquanto há a comodidade de ter vários programas prontos disponíveis, no final precisamos de mais recursos do que essa Máquina Virtual foi capaz de nos fornecer. Acabamos apenas instalando todos os recursos necessários no Linux padrão mesmo.

Note que é importante você instalar os adicionais para convidado também. Certifique-se de utilizar a versão mais recente da máquina virtual, e adicionar as extensões para se utilizar usb

Instalando PYTHON

Quase todos os programas do crazyflie exigem os recursos do Python, logo essa deve ser a primeira coisa que se deve fazer. Você pode utilizar o comando abaixo, ou algum outro comando. É recomendável que use python3, pois diversos recursos só são disponíveis lá. É importante instalar o pip3 também

- `sudo apt install python3`
- `sudo apt install python3-pip`

Instalando Git

É importante também que você tenha uma conta no github, visto que muitos dos programas são puxados por lá. Certifique-se também que o git esteja instalado na sua máquina:

- `sudo apt install git`

Um próximo passo para poder fazer todos os comandos do git é permitir o acesso SSH da sua máquina no seu repositório do git. Para isso, siga o tutorial descrito [aqui](#).

Você precisa

- Verificar se você já criou uma SSH key
- Criar uma SSH key
- Adicionar essa SSH key para a sua ssh key agent
- Adicionar essa SSH key na sua conta git

Todos esses passos estão bem encaminhados no tutorial do github.

Instalando cflib

A biblioteca cflib é a biblioteca básica para se controlar o Crazyflie, e deve ser a primeira biblioteca a ser instalada. Você pode encontrar a última versão no github (<https://github.com/bitcraze/crazyflie-lib-python>). Siga as instruções no read-me, ou seja, clone, e depois instale a biblioteca.

- De um fork da biblioteca cflib para a sua conta local
- Clone para o seu repositório local
 - `git clone git@github.com:USERNAME/crazyflie-lib-python.git`
- Instale a biblioteca na sua máquina
 - `pip3 install -e ~/crazyflie-lib-python`

Para testar se deu certo, uma maneira é, a partir do terminal:

- `python3`
- `import cflib`

Instalando cfclient

O cfclient é uma interface de controle gráfico do crazyflie, e é muito útil para poder extrair alguns parâmetros do mini-drone, como a altura, a calibragem, e outros. A instalação dele também é simples, semelhante ao cflib. Você deve encontrar a última versão no github (<https://github.com/bitcraze/crazyflie-clients-python>). Siga as instruções do read-me

- uma das coisas que ele pede é instalar alguns outros recursos do python. Por exemplo:
 - `sudo apt-get install python3 python3-pip python3-pyqt5 python3-pyqt5.qtsvg`
- Dê um fork da biblioteca cfclient para a sua conta local
- Clone para o seu repositório local
 - `git clone git@github.com:USERNAME/crazyflie-clients-python`
- entre na pasta clonada, e instale o programa
 - `sudo pip3 install -e .`
- Para abrir o programa, teclle cfclient
 - Se Você não tiver alterado o programa para receber o CrazyRadio, você precisará fazer `sudo`

Instalando OpenCV

Para podermos fazer a navegação autônoma do mini-drone, nós utilizamos a biblioteca do openCV. É um pouco complicado e demorado fazer a instalação disso, por isso tenha paciência

- Instale algumas bibliotecas auxiliares
 - `pip3 install matplotlib`
 - `pip3 install numpy`
 - `sudo apt install python-dev`
 - `sudo apt-get install python3-tk`
- Instale o opencv e os módulos extras
 - `pip3 install opencv-python`
 - `pip3 install opencv-contrib-python`

Talvez você precise disso

- *Clone o repositório do opencv*
 - `git clone https://github.com/opencv/opencv.git`
- *Instalar algumas dependencias adicionais (voce pode ou pode não precisar disso)*
 - `sudo apt install libgtk-3-dev`
 - `sudo apt install ubuntu-restricted-extras`
 - `sudo apt install libgstreamer-plugins-base1.0-dev`
 - `sudo apt install python`
 - `sudo apt-get install libatlas-base-dev`

CIAB

A CIAB é uma feira anual de automação bancária, que a Scopus tem a oportunidade de demonstrar alguns dos seus projetos para o mercado. No caso do mini-drone, foi mostrado uma pequena demonstração de um movimento autônomo, transmitido por uma câmera instalada no FabLab para a feira. O usuário podia, no site, escolher um entre três movimentos, e quando ele clicava, o site enviava um comando no servidor. No fabLab, havia um programa que puxava do servidor esse comando, e executava esse comando no drone, que podia ser visto da feira. Tentaremos explicar esse fluxo a seguir.

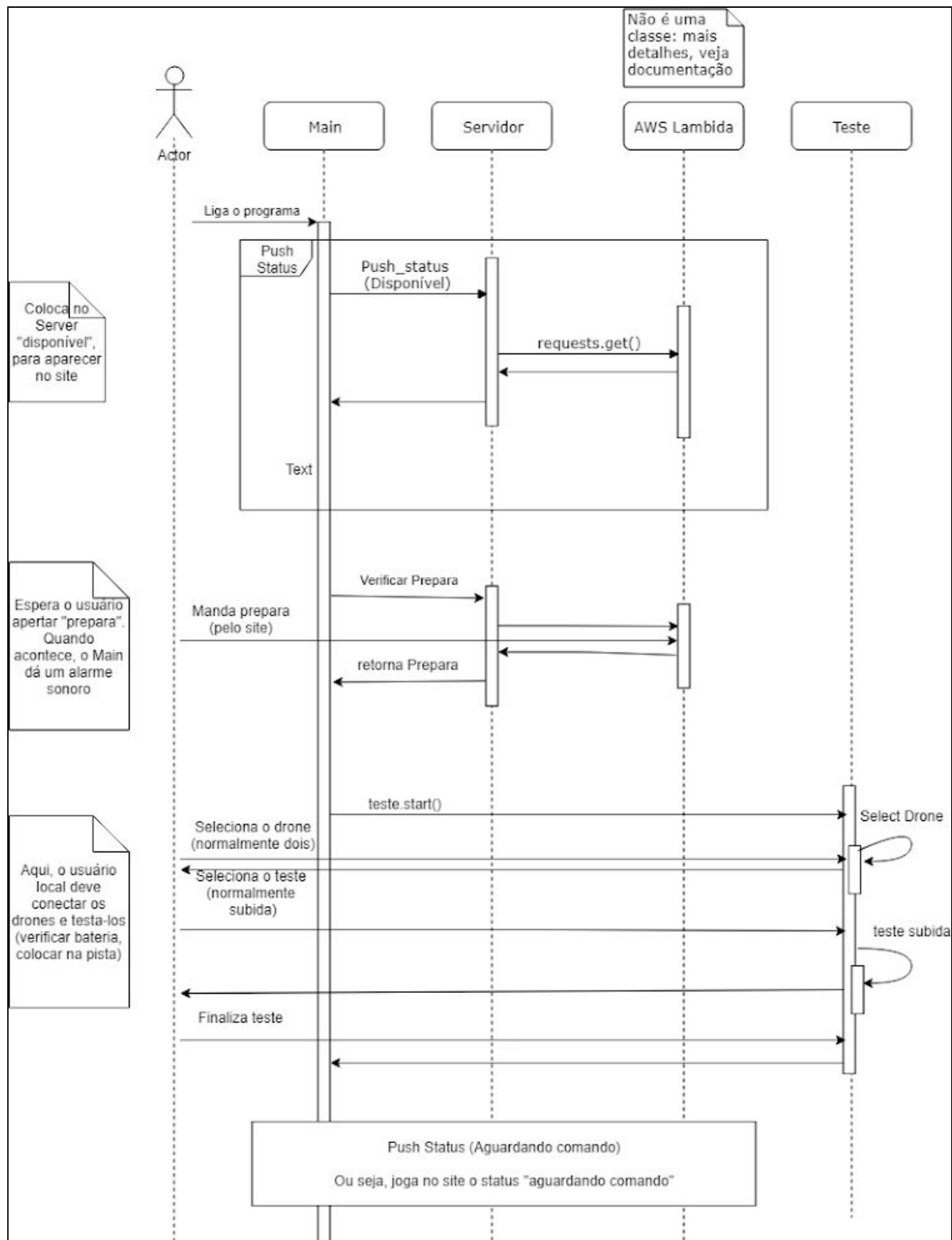
Ações necessárias

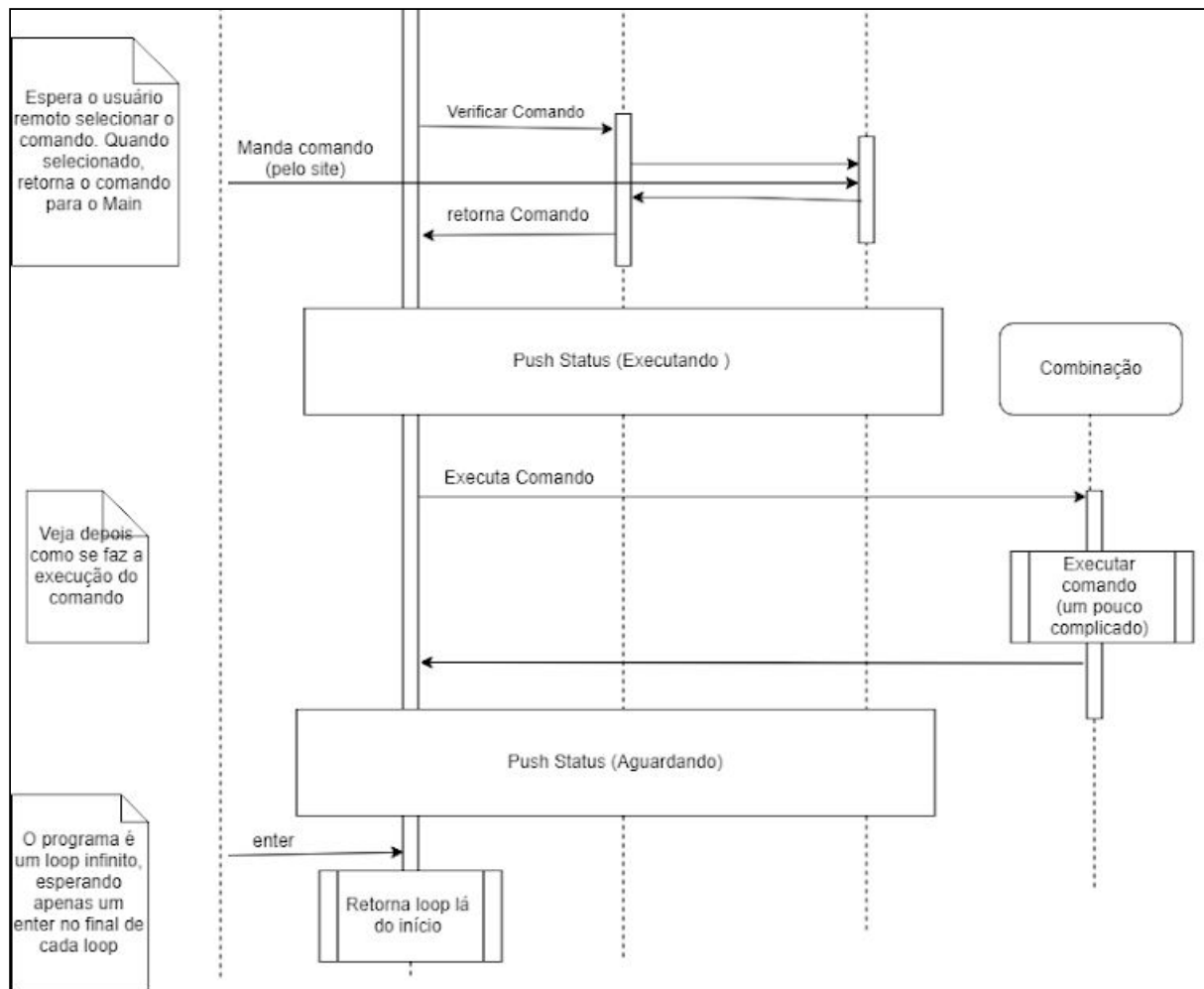
- Para controlar a entrada de 'enter' no programa
 - pip3 install pynput
- Para fazer http requests
 - pip3 install requests
- Para fazer som de alarme
 - sudo apt install sox
- Substituir o motion_commander.py do cflib (cflib/positioning) pelo motion_commander.py nosso

Resumo da sequência

- 1) Ligar o programa
- 2) O cliente manda um comando "prepara", através do site, passando pelo servidor
- 3) O programa busca esse comando no servidor, e dá um alarme sonoro
- 4) O usuário prepara os drones, faz os testes necessários, coloca na pista, e aguarda o comando da trajetória
- 5) O cliente manda uma trajetória, novamente através do site
- 6) O programa busca essa trajetória no servidor, e manda os drones realizarem essa trajetória
- 7) Feita a trajetória, o usuário tira os drones da pista, faz as manutenções necessárias, e aguarda o comando prepara, voltando ao passo 2.

Um diagrama entre as classes pode ser visto no png anexo





Como funciona o servidor

Foi utilizada como o servidor o serviço AWS Lambda, integrada com uma API WEB. De uma maneira simplificada, o servidor serve como uma pilha de tamanho um: você pode enviar um comando nela (PUSH), e ela irá armazenar esse comando. Para consumir essa informação, você pode dar um POP ou um PEEK: o primeiro, lendo e deletando o dado da pilha, e o segundo, somente lendo o dado da pilha sem retirá-la de lá. Se você der um PUSH mesmo com um dado na pilha, isso irá substituir o dado presente, ou seja, vale sempre o último dado escrito. O código do Lambda está disponível no git.

Modo de utilizar o servidor:

Ele funciona analogamente à uma pilha de tamanho 1, implementando as funções tradicionais: POP, PUSH e PEEK.

- PUSH: Para adicionar um comando ("linear") da pilha do drone ("d1") devemos invocar:

<https://3vkeycenej.execute-api.us-east-1.amazonaws.com/prod/CIAB-2018-DroneComm andQueue?droneId=d1&operation=PUSH&command=linear&secret=dr0neRulez2A5T7U>

Se o método for bem-sucedido, será retornado o texto "OK"

- POP: Para obter um comando da pilha do drone ("d1") devemos invocar:
[https://3vkeycenej.execute-api.us-east-1.amazonaws.com/prod/CIAB-2018-DroneComm
andQueue?droneId=1&operation=POP&secret=dr0neRulez2A5T7U](https://3vkeycenej.execute-api.us-east-1.amazonaws.com/prod/CIAB-2018-DroneCommandQueue?droneId=1&operation=POP&secret=dr0neRulez2A5T7U)

Será retornado o comando. Caso não tenha comando algum na fila, será retornado "empty"

- PEEK: Para visualizar um comando da pilha do drone ("d1"), sem removê-lo da pilha devemos invocar:

[https://3vkeycenej.execute-api.us-east-1.amazonaws.com/prod/CIAB-2018-DroneComm
andQueue?droneId=1&operation=PEEK&secret=dr0neRulez2A5T7U](https://3vkeycenej.execute-api.us-east-1.amazonaws.com/prod/CIAB-2018-DroneCommandQueue?droneId=1&operation=PEEK&secret=dr0neRulez2A5T7U)

Será retornado o comando. Caso não tenha comando algum na fila, será retornado "empty"

Como funciona o site

O site depende do servidor para funcionar, e tem também um link para três vídeos no youtube, para ser utilizado caso os movimentos do drone gerem algum problema.

<https://s3.amazonaws.com/ciab2018-minidrone/index.html>

O site foi projetado para funcionar na televisão da Ciab, então pode não ficar muito responsivo para alguns aparelhos



Algumas dicas úteis

- É aconselhável deixar o endereço URL dos mini drones em endereços diferentes. A interferência entre eles será diminuída, e o movimento dos drones será mais seguro e mais robusto
- Foi tentado tirar o filtro de Kalman, para ver se a estabilidade dele melhorava. Não foi possível chegar a uma conclusão definitiva, mas parece que os movimentos pioravam quando não há o filtro de Kalman