

Controlador de Nó de Comunicações Voador Sensível ao Tráfego

Estágio de verão CTM 2020

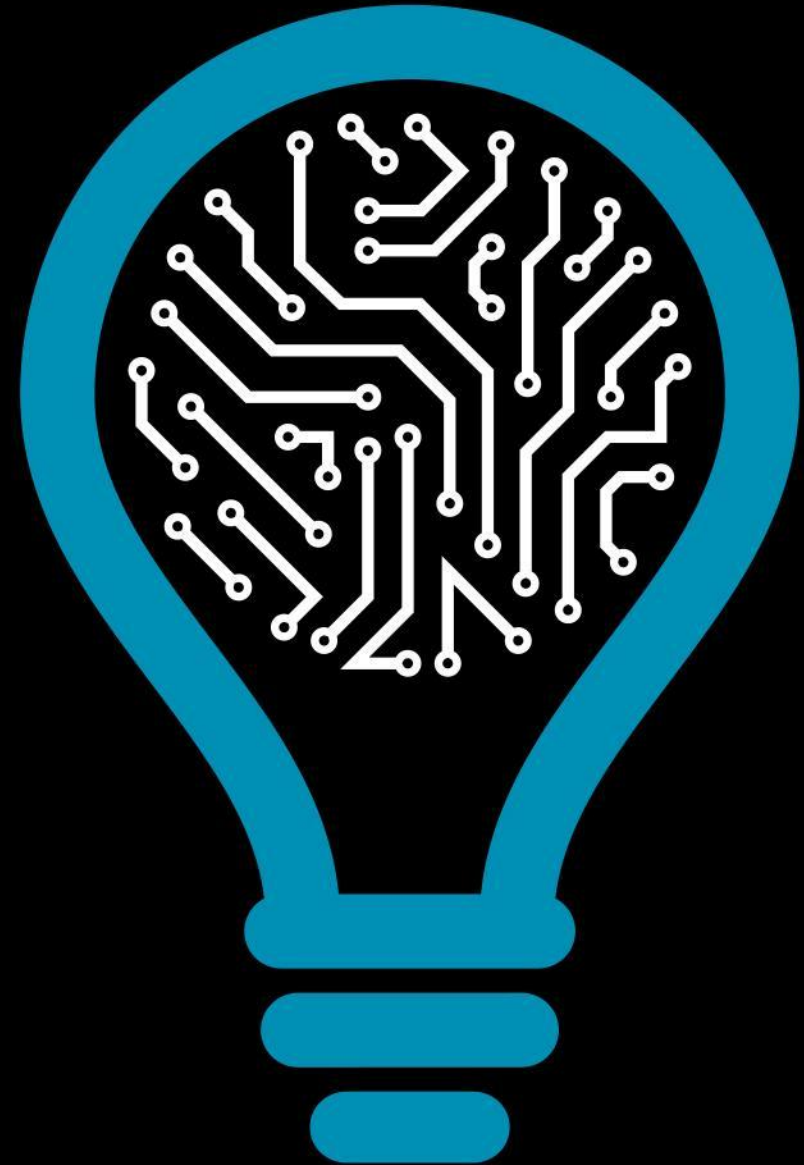
Orientador: André Filipe Coelho

Eduardo da Costa Correia
João Carlos Carreira Martins

31 de julho de 2020



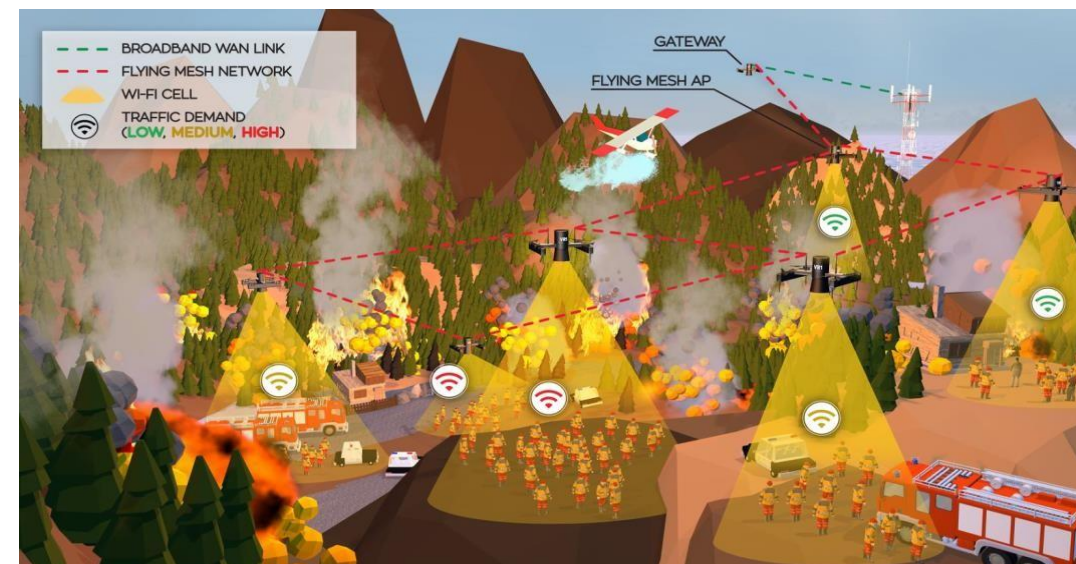
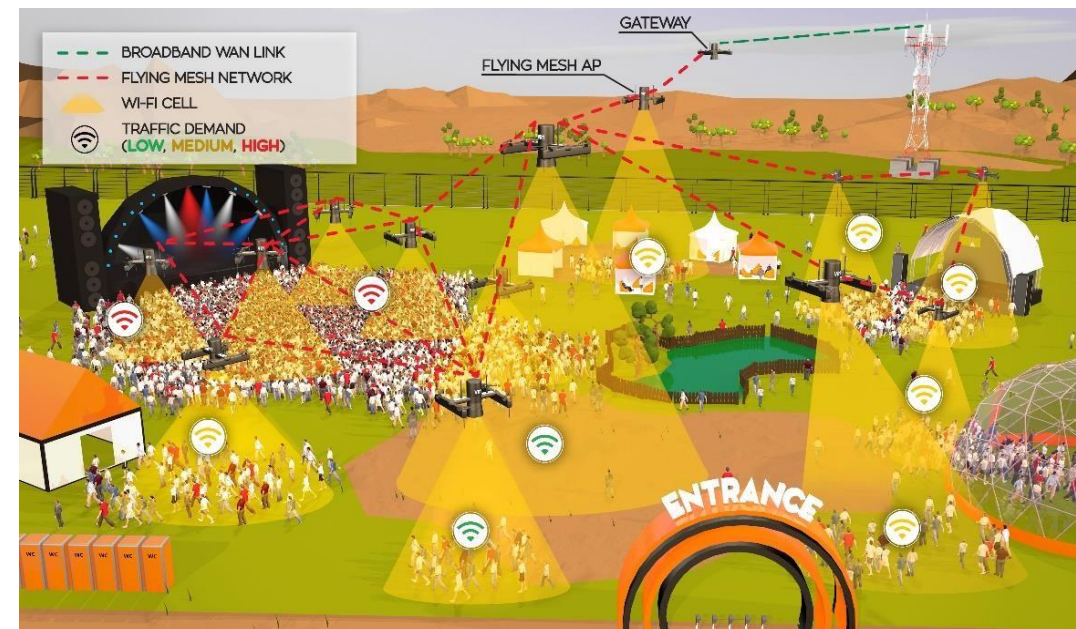
INSTITUTO DE ENGENHARIA
DE SISTEMAS E COMPUTADORES,
TECNOLOGIA E CIÊNCIA





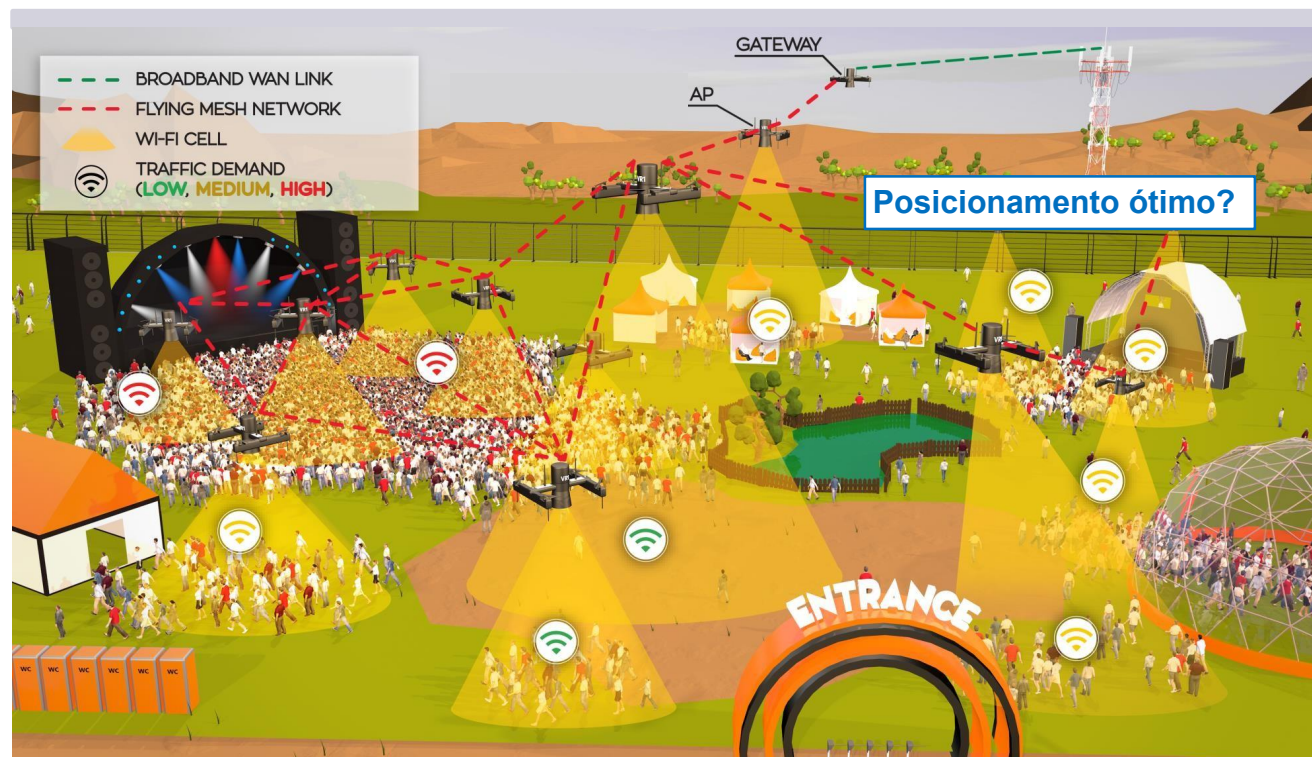
Enquadramento

- **Eventos temporários sobrelotados**
 - Alta densidade de dispositivos móveis
 - Tráfego variável ao longo do tempo
 - Acesso à Internet de banda-larga
- **Veículos Aéreos Não Tripulados (UAVs)**
 - Operam virtualmente em qualquer lugar
 - Pairam sobre o solo e transportam carga
 - **Excelentes plataformas para nós de comunicações**
- **Redes Voadoras**
 - Pontos de Acesso (APs) Wi-Fi e células 4G/5G
 - Estabelecimento e reforço da cobertura s/ fios



Objetivos

1. **Implementação** de **software** para **cálculo da posição ótima** do UAV
2. **Definição da posição** ótima calculada **no UAV**
3. **Validação e avaliação** da solução desenvolvida em ambiente de simulação



Ferramentas utilizadas

- **Python:** linguagem de programação usada
- **Gekko:** resolução do sistema de inequações do algoritmo GWP
- **ArduPilot SITL:** simulação de um veículo aéreo não tripulado (UAV)
- **DroneKit-Python API:** comunicação com o controlador de voo ArduPilot
- **Folium:** representação do cenário num mapa interativo



Algoritmo GWP

```

1:  $P_T = 0$ 
2: while true do
3:   Solve system of equations
4:   if  $(x, y, z)_{UAV\ relay} \neq \emptyset$  then:
5:     return  $P_T, (x, y, z)_{relay}$ 
6:   else
7:      $P_T = P_T + 1$ 
8:   end if
9: end while

```

- Cálculo da posição ótima do UAV Relay, correspondente à interseção das esferas que representam a zona de alcance de cada FAP.
- No nosso caso, abordámos a resolução do sistema de inequações como um problema de otimização o qual foi resolvido através da utilização do GEKKO

$$\begin{cases}
 (x_{UAV\ relay} - x_{FAP\ 1})^2 + (y_{UAV\ relay} - y_{FAP\ 1})^2 + (z_{UAV\ relay} - z_{FAP\ 1})^2 \leq \left(10^{\frac{K + P_T - SNR_{Ligação\ 1}}{20}}\right)^2 \\
 \dots \\
 (x_{UAV\ relay} - x_{FAP\ N})^2 + (y_{UAV\ relay} - y_{FAP\ N})^2 + (z_{UAV\ relay} - z_{FAP\ N})^2 \leq \left(10^{\frac{K + P_T - SNR_{Ligação\ N}}{20}}\right)^2
 \end{cases}$$

$$K = -20 \log_{10} \left(\frac{4\pi}{3 \times 10^8} \right) - 20 \log_{10} (5250 \times 10^6) - (-85)$$

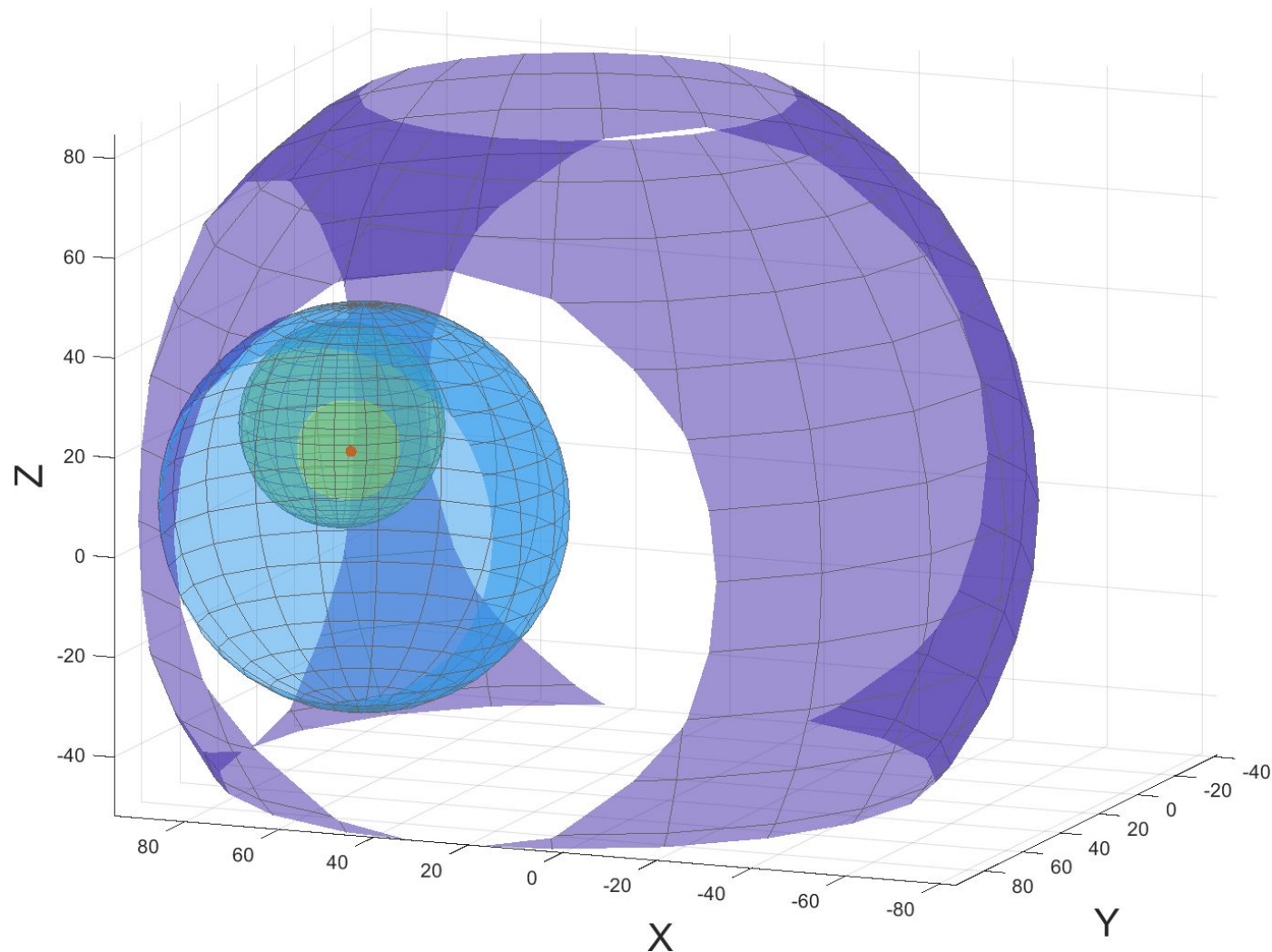
Trabalho desenvolvido

- **UAV_simulation:**
 - Método desenvolvido que simula um cenário real em que os pontos de acesso (FAPs) enviam a sua posição e o tráfego oferecido ao UAV, que atualiza a sua posição de acordo com estes dados.
 - **Etapas:**
 - Decolagem do UAV
 - Para cada vez que os FAPs enviem informação ao UAV:
 - Cálculo da posição ótima do UAV com o algoritmo **GWP**
 - Mover o UAV para a posição ótima
 - Representação da solução num mapa interativo usando **Folium**¹
 - Aterragem do UAV no ponto de partida

¹ Mapa interativo, gerado sempre que é lido um novo ficheiro de input pela simulação e que representa toda a informação que não é possível apresentar no SITL.

Resultados obtidos

Representação inicial dos FAPs e o seu alcance (centro das esferas e o seu raio) bem como da posição ótima do UAV Relay (ponto vermelho) após executar o algoritmo GWP.



Resultados obtidos

Representação em texto da solução do algoritmo **GWP**, que calcula a posição ótima do UAV.

- **PT**: Power Transmission
- **Location**: coordenadas x, y, z, em metros, da posição ótima do UAV. As coordenadas x e y são posteriormente convertidas em latitude e longitude, respetivamente, para serem utilizadas no **ArduPilot**.

```
Press ENTER to get new FAPs information (filename FAPs Info/input_1.csv):  
  
PT: 12  
Location:  
  x: 58.420756776  
  y: 35.360059176  
  z: 9.4887323672  
Set default/target airspeed to 10  
Going towards the optimal position ...
```




Resultados obtidos

Simulação do voo em ambiente SITL.

Validação da simulação acima num mapa interativo em que estão representados o UAV e os diversos FAPs com o seu alcance sem-fios.



Conclusões

- Os objetivos em mente foram todos cumpridos, incluindo a aplicação e validação do algoritmo GWP bem como a simulação do voo do UAV.
- Apesar deste trabalho ter sido realizado na íntegra em ambiente virtual, foi possível obter uma previsão precisa do comportamento do drone, bem como validar todo o algoritmo para ser aplicado numa situação real.
- Com o SITL não se verificam, no entanto, colisões que existiriam num cenário real.