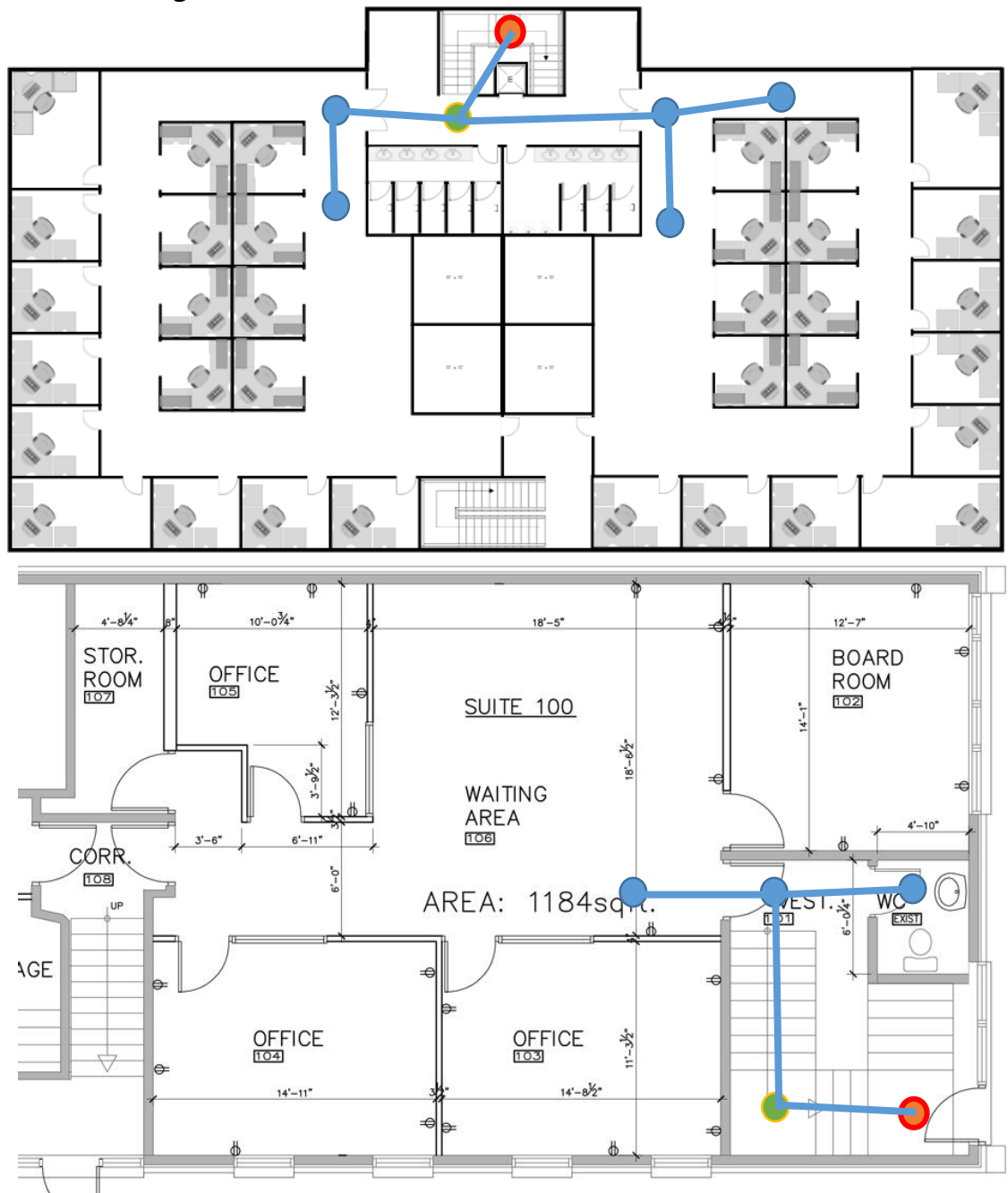


## FOR DISCUSSION

### ALGORITHM TO NAVIGATE BETWEEN FLOORS

The algorithm is based on an assumption that the navigation on the floors happens as described in proximity navigation algorithm, i.e. User follows the graph edges.

1. Before the real-time navigation, the navigation system forms the so-called “transfer edges”. The transfer edge consists of one common vertex (red beacon) and at least two edges (edge formed with red and green beacons) that belong to different floors. Possible combinations are shown next



2. The real-time part works as follows:

When the User approaches the common beacon (i.e. the red beacon sends the highest RSSI), the new map must become ready for being shown to the User.

When the determined User position reaches or breaks the red beacon, then algorithm switches the map and continues the navigation.

If the User approaches the red beacon, but does not break it and turns back, then the map remains the same.

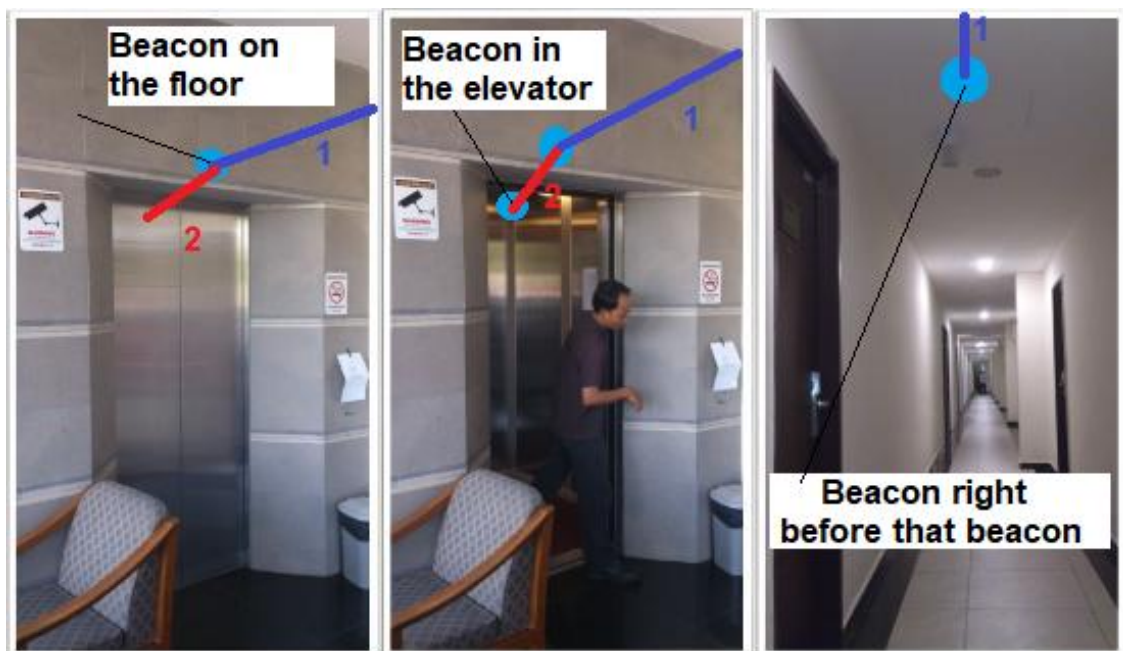
*I expect the distance limiter to work very well for this case, but its performance must be tested.*

### Elevator case

This part of the proposition deals with an algorithm of how we expect to detect the User entering the elevator and leaving the latest.

User entering the elevator:

When the User approaches the elevator, the beacon installed in the elevator sends the most powerful RSSI signal. At this moment the application repositions the User right to the elevator (merges the coordinates of the User with the coordinates of beacon in the elevator). The map freezes and stays frozen until the escape-from-elevator event is detected.



Then the User goes up in the elevator and passes some floors.

Handling of the probable cases:

**The elevator stops, but the User does not leave the latest.** The application starts receiving the BLE packages from the beacon marked as “beacon on the floor” in the figure. Meanwhile the IMU part checks the steps and buffers three steps. If, after this, the accelerometer detects the abnormal acceleration (the elevator starts lifting or descending), then the User is again put into the elevator and the buffer is cleared. The map is not loaded.

**The elevator stops, User leaves the elevator, but then gets back.** The application starts receiving the BLE packages from the beacon marked as “beacon on the floor” in the figure. Meanwhile the IMU part checks the steps and buffers three steps. After this, no abnormal acceleration is observed. Then the Application loads the map and shows User position considering the three steps from the buffer. Then standard navigation starts. If the User enters back, then the handling of this event happens as described above.

**The User leaves the elevator.** The sequence is similar to the previous case.