Cleaned up GitHub files

Fixing GUI

Adding new algorithm

# NEW Algorithm

## Current idea and method

**Two ideas for when to optimize**

1. Individual KPIs (calculated before weighing KPIs)
   1. Each KPI has a unique threshold set in the simulator GUI that is compared with the corresponding user’s UE Record value for that KPI. If the user’s reported value is below the set threshold, optimization will occur.
   2. By setting the KPI threshold to below minimum value in range, optimization will always occur since the reported user value is greater than threshold value.

Example: User RSRP value: -74 dBm, Threshold RSRP value: -50 dBm => Optimize

1. General KPI (calculated after weighing all KPIs)
   1. Every individual KPI is given a weighted percentage value (totaling to 100 percent) that is assigned to a general/overall KPI threshold. Likewise, each user’s KPI values are put through the weighing algorithm and compared to the threshold. If the user’s general KPI is below the threshold, then optimization will occur for that user.

**Individual vs. General**

* Individual provides more fine tuning of the network to ensure that each user is provided a higher network quality.
  + Compares values for each KPI, if lower optimizes
  + Could be prone to switching base stations more depending on various KPI values
  + Need individual thresholds for each KPI value to compare to since the unit and ranges for each value is different depending on the KPI.
* General will switch users less maintaining the current connections if a predetermined quality is met.
  + In some cases, if the weight of certain KPIs is low, the user may experience less than optimal service if general value is met (does not account for individual KPIs)
  + One threshold value due to the KPIs being weighted and combined to calculate a single KPI value.

## Pseudocode (Green = Done, Orange/Red = Not Done)

|  |
| --- |
| Make boolean doneHealing to separate healing from optimization  Once healing is complete, then optimize afterwards (set doneHealing to true) |
| Healing  Get failing BaseStations and congested BaseStations  Start with failing BSs   * Offload users to closest BS based on distance * Highest demand users should go first |
| Optimization (Individual)  Get BaseStations  Get UEDB from BaseStation  Read UEDB and create vector pair of UserID and specific KPI to be measured against threshold  If user value of chosen KPI is abo its threshold value, do not optimize  Otherwise, optimize according to which KPI is below the threshold value  **WIP**  **RSRP (Received power is bad, move to a closer BaseStation)**   * Call offload function * Check status of each nearby BaseStation * Move user to the closest distance BaseStation   **Not implemented yet**  **DDR (Do something about fixing DataDropRate)**  **RSRQ (Quality of signal is bad, do something about it)**  **RSSI (Strength of signal is bad, move to a closer BaseStation?)** |
| Optimization (General)  Get KPI based on which threshold   * RSRP (Reference Signal Received Power) * DDR (Data Drop Rate) * ~~RSRQ (Reference Signal Received Quality)~~[[1]](#footnote-2) * ~~RSSI (Reference Signal Strength Index)~~[[2]](#footnote-3)   Weigh KPIs   * Only RSRP and DDR * Based on percentage (importance)   Determine what BaseStation to offload to   * Basestation status (priority) * Based on the weight KPIs (secondary)   Call offload function |

## Future Implementations

### Algorithm

Complete individual KPI code

### GUI

Adding slider to adjust KPIs (for optimization (general))

* When KPI weight is above 100% when combined, adjust the other sliders to match?

Adding toggle option in GUI to choose between using general KPI optimization or individual KPI optimization

1. Not used because they are already proportional to RSRP [↑](#footnote-ref-2)
2. Same as (1) [↑](#footnote-ref-3)