o

Self-Healing 5G Network

DESIGN DOCUMENT

Team 36
Prof.Selim
Matthew Johnson, Amber Chin, Parnumart Hanthiradej,Adam Tiedman, Yuin-Choon Ng
Team Email: sdmay21-36@iastate.edu
Team Website: https://sdmay21-36.sd.ece.iastate.edu

Revised: 10/4/20|Version 1

Executive Summary

Development Standards & Practices Used

- Software Development Standards
  - Python Coding Experience to design and implement algorithms
  - Linux Proficiency to operate testbed servers
- Networking Knowledge
  - Knowledge of wireless protocols
    - 5g
    - WiFi
    - Bluetooth
    - UDP/TCP
  - Knowledge of wireless nodes
    - WLAN
    - Radio Nodes
    - Support Servers
  - Knowledge of network testbeds
    - GENI(Wired)
    - ORBIT(Wireless)

Summary of Requirements

- The network needs to be able to:
  - Detect an outage
  - Diagnose the outage
  - Compensate for the damage while attempting a fix
  - Perform these above steps within a time efficient manner(milliseconds)
- The network should be:
  - Compatible with common network nodes
  - Modifiable to suit specific network needs
  - Usable for most 5g applications

Applicable Courses from Iowa State University Curriculum

CprE 489

CprE 308

New Skills/Knowledge acquired that was not taught in courses

- 5G network protocols
- Self-healing algorithm design

Table of Contents

1 Introduction 4

1.1 5

1.2 5

1.3 5

1.4 5

1.5 5

1.6 6

1.7 6

2 6

2.1 Task Decomposition 5

2.2 Risks And Risk Management/Mitigation 6

2.3 Project Proposed Milestones, Metrics, and Evaluation Criteria 6

2.4 Project Timeline/Schedule 6

2.5 Project Tracking Procedures 6

2.6 Personnel Effort Requirements 7

2.7 Other Resource Requirements 7

2.8 Financial Requirements 7

3 Design 7

3.1 Previous Work And Literature 7

3.2 10

3.3 10

3.4 Technology Considerations 8

3.5 Design Analysis 8

3.6 11

3.7 11

4 Testing 9

4.1 12

4.2 12

4.3 12

4.4 12

5 Implementation 10

6 Closing Material 10
6.1 Conclusion 10
6.2 References 10
6.3 Appendices 10

List of figures/tables/symbols/definitions (This should be the similar to the project plan)

1 Introduction

1.1 Acknowledgement

Our current advisor is Professor Selim we would like to thank him for helping guide us through the early stages of our project. He has provided us with large amounts of background knowledge as well as helped to guide our research and overall project.

1.2 Problem and Project Statement

Current cellular networks will portion out a large amount of their budget to spend on repairing and resolving network problems and outages. The current method of resolution is to rely on human expertise to identify, diagnose, and resolve any issues with the network. This process has proven to not only be very costly, but also presents significant challenges in regards to the speed at which resolutions can be made. With 5G networks on their way the complexity and overall cell density will prove to be too much to handle with the current processes. To be able to keep next generation networks running a new solution for resolving network issues must be created.

The proposed solution is to create a self healing network that utilizes automation to repair damaged networks. The system can utilize machine learning to detect a full or partial outage, diagnose the cause, and compensate/fix the problem. Using an algorithm to solve these issues will allow for a network to have minimal damage during an outage while keeping the cost to maintain the network much lower than in previous generations.

1.3 Operational Environment

To fully test the self healing network a real world simulation is needed. For the ORBIT platform that means utilizing the outdoor testing environment. The outdoor environment will allow for testing of unexpected variables such as radio noise, propagation issues, and any other variables seen by cellular towers.

1.4 Requirements

For a self healing network there are three main requirements to be fulfilled. The network needs to be able to detect an outage, diagnose the outage, and compensate for the damage while attempting a fix. Although there are no “hard requirements” the goal of the project is to improve upon the current generation systems. Ultimately a perfect solution will leave the end user unaware that there were any issues with the network at all while being able to perform at a much lower cost of operation.

1.5 Intended Users and Uses

5G is designed to do a variety of tasks which improve our lives. For example, fast download speeds, low latency, and billions of connectivity to devices such as virtual reality and artificial intelligence.

Self healing 5G network is used when there is a failure in a network. The self healing 5G network can resolve the problem without humans involved. It is a tool which can detect, remediate outages, failure and breaches.

1.6 Assumptions and Limitations

Assumptions:

- The end users of the neighbouring base stations will not be affected by base station failure.
- The project should be working in an Indoor Radio Grid Testbed and across large-scale open-access wireless networks.
- There will be no presence of interference issues or network traffic.
- The algorithm of the project must have the functionality of being able to decide which base station can be involved in the healing process.

Limitations:

- The financial cost for this project has to be none, since we are using free platforms to conduct experiments on this project.
- The software development standards for this project should be restricted to using python for coding to implement the self-healing algorithms on the testbeds and linux to operate the testbed servers.

1.7 Expected End Product and Deliverables

Expected End product in the first semester is to implement a simple self-healing algorithm between two access points without involving any users.

Expected deliverables of this project are writing and preparing to submit a scientific paper to one of the IEEE conferences or workshops. Also one final presentation of the project within one week before the end of first semester.

2 Project Plan

2.1 Task Decomposition

- Research Orbit Platform
  - Begin testing on simulation platform Geni
  - Run test scripts on real Orbit platform devices
- Research Self-Healing Algorithms
  - Do a research search for important self-healing algorithm descriptions
  - Break these topic down into important concepts
  - Choose desired algorithm and map out key block diagram

- Implement a self-healing algorithm on Orbit platform
    - Code algorithm in Python
    - Implement algorithm on two access points
    - Implement algorithm on real-world outdoor test network
- Assess self-healing algorithm
    - Assess quality of chosen algorithm implementation
    - Decide if this algorithm meets our desired benchmarks
    - Make necessary changes to algorithm and retest
- Project Presentation
    - Prepare presentation on outcome of self-healing algorithm implementation
    - Prepare paper for IEEE conference
    - Give presentation

2.2 Risks And Risk Management/Mitigation

Implement self-healing algorithm on Orbit platform:

The Orbit platform may not work well for our implementation of a self-healing algorithm. Because this platform is well known and used for applications such as ours, the probability of this is fairly low at around .2. If this is the case we may need to select a different algorithm, or we may need to choose a different platform such as Geni.

Assess self-healing algorithm

The second risk factor is that our algorithm may not meet our expectations. This is fairly likely at about .6. If this is the case, then our plan will be to go back to the research stage and choose an algorithm with the knowledge gained through our first implementation. We would then need to go through the implementation task again, finally reaching the assess task once more. This risk can be slightly mitigated through a more lengthy research process, however, it can never be eliminated until we actually test the algorithm on the platform. We have factored into our timeline that this may occur, so the loss in time is not drastic.

2.3 Project Proposed Milestones, Metrics, and Evaluation Criteria

Research Orbit Platform :

The research on Orbit platform has been carried out. It takes 2 to 3 weeks to receive a response from the organization. Because of this, our team has planned to use another platform called GENI which is much similar to the Orbit platform. On GENI, team members are allowed to create their own slices (experiments) with different nodes (server or client).

Research about what self-healing is:

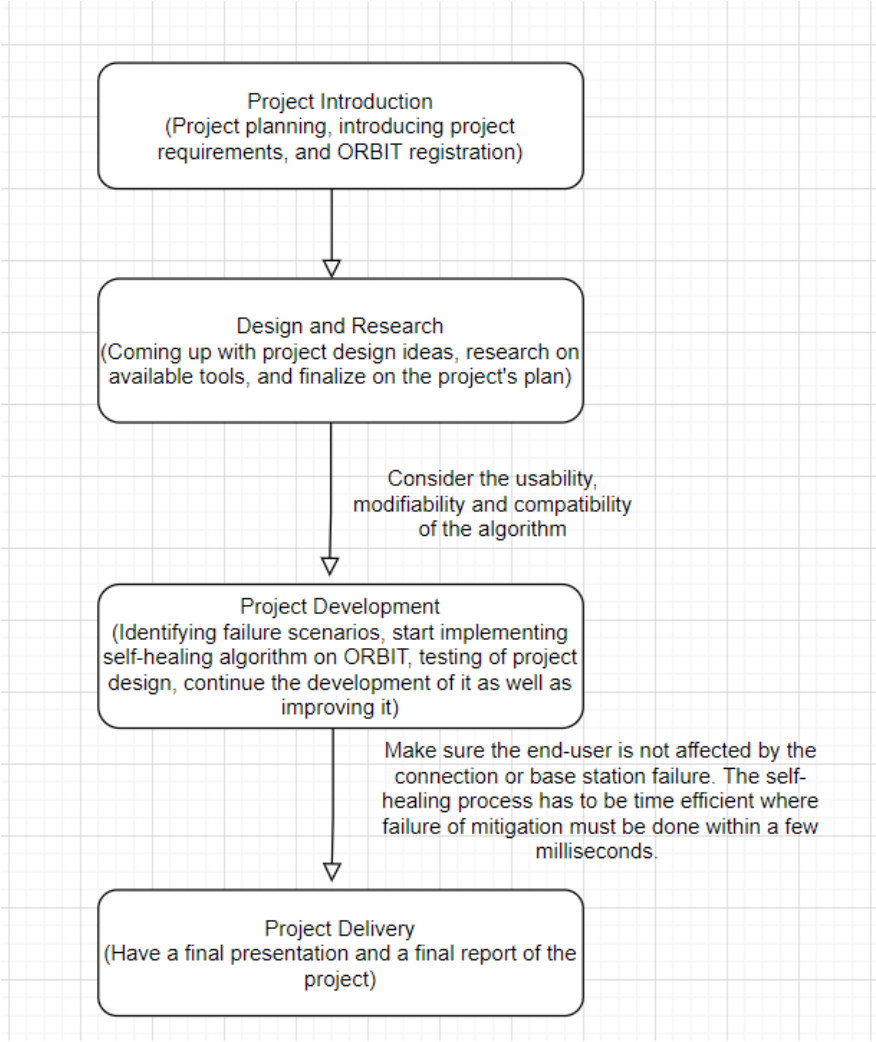
Our team has been doing a lot of research and reading IEEE articles on what self-healing is and also presenting powerpoint slides to our advisor to get feedback or additional information from him biweekly. We learnt that the most important thing about self-healing is that the end user should

not have to experience any network failure from the base stations. Self-healing is a fundamental process in any network.

Research Self-Healing algorithm :

The research on Self-Healing Algorithms has been carried out. On the 5G network, our team found out that the most failure in the connection of the 5G network is because the high frequency signal cannot travel through obstacles and it can be absorbed by plants and buildings.. Because of this, our team will be working on an algorithm on how to connect to devices which lose connection to the 5G network.

2.4 Project Timeline/Schedule



The flowchart portrays the project's timeline for two semesters. First and foremost, we are going to start off with a project introduction where we learn what requirements are needed in terms of

skills, knowledge and experience that we can contribute to this project. We also need to register for accounts in the free platforms which are GENI and the ORBIT testbed. In this stage, project planning is an essential task as it ensures that we are on track in this project.

After this, we will proceed with designing and doing a lot of research on the project. We will have to come up with project design ideas, do research on the available resources to make this project successful and then finalize on the project's plan. The project design ideas are crucial because it has to include the project's functional requirements which are the effects of a network outage on network users should be mitigated and the self-healing of the network should be time efficient where the failure mitigation must be done within a few milliseconds. Other than that, usability, modifiability and compatibility of the algorithm has to be ensured. The algorithm should be usable for most 5G network applications, modifiable to suit specific network needs and compatible with common network node hardwares.

Next is the program development where we start implementing the self-healing algorithm on the ORBIT testbed, identifying failure scenarios through experiments. testing the project's design and continuing the development of it as well as improving it. This stage is fundamental in the process of the project because this is where the most progress can be viewed and certain adjustments or modifications can be made in our project. The testing for this project would include both virtual and real world tests. This stage is also where we can fix mistakes and enhance the algorithm's efficiency and functionality. We have to make sure that the end-user is not affected by the connection or base station failure. The self-healing process has to be time efficient where the mitigation failure must be done within a few milliseconds.

Last but not least is the project delivery. This would be done at the end of the second semester where we would have a final presentation and a final report of the project.

2.5 Project Tracking Procedures

We're using the GENI portal created by the professor to keep track of our Self-healing-5G project. In the GENI portal, we can decide who will be the project leader, leading other members in presenting the project status to the professor each week. All members can create slices for project explanation and also be able to check the update of the project's process.

2.6 Personnel Effort Requirements

Individuals need to have a basic knowledge of coding on Linux commands and Python to implement Self-Healing algorithms on testbed and operate testbed servers. During the first semester, all members need to learn how to use GENI experiment network deployment because it serves as a proof of concept experiment platform for wireless aspects.

2.7 Other Resource Requirements

1.Radio Node is a Primary platform for user experiments controlled by a chassis manager through an ethernet interface to reset power on/off.

2. Instrumentation subsystem is a critical component of an automation system create to measures radio signal levels and creates artificial RF interference

3. WLAN monitor system provides a network layer view of the grid's components using several WLAN "observers" spread across the system

4. Support Servers includes front-end servers for web services and backend for experimentation and data storage

2.8 Financial Requirements

There are no financial requirements for the project because we will be using free platforms such as GENI and ORBIT. Experiments will be conducted with those platforms and coding will be done mostly in Python and Linux.

3 Design

3.1 Previous Work And Literature

Include relevant background/literature review for the project

– If similar products exist in the market, describe what has already been done

– If you are following previous work, cite that and discuss the **advantages/shortcomings**

– Note that while you are not expected to "compete" with other existing products / research groups, you should be able to differentiate your project from what is available

Detail any similar products or research done on this topic previously. Please cite your sources and include them in your references. All figures must be captioned and referenced in your text.

3.2 Design Thinking

Detail any design thinking driven design "define" aspects that shape your design. Enumerate some of the other design choices that came up in your design thinking "ideate" phase.

3.3 Proposed Design

Include any/all possible methods of approach to solving the problem:

- Discuss what you have done so far – what have you tried/implemented/tested?
- Some discussion of how this design satisfies the **functional and non-functional requirements** of the project.
- If any **standards** are relevant to your project (e.g. IEEE standards, NIST standards) discuss the applicability of those standards here
- This design description should be in **sufficient detail** that another team of engineers can look through it and implement it.

3.4 Technology Considerations

Highlight the strengths, weakness, and trade-offs made in technology available.

Discuss possible solutions and design alternatives

3.5 Design Analysis

- Did your proposed design from 3.3 work? Why or why not?
- What are your observations, thoughts, and ideas to modify or iterate over the design?

3.6 Development Process

Discuss what development process you are following with a rationale for it – Waterfall, TDD, Agile. Note that this is not necessarily only for software projects. Development processes are applicable for all design projects.

3.7 Design Plan

Describe a design plan with respect to use-cases within the context of requirements, modules in your design (dependency/concurrency of modules through a module diagram, interfaces, architectural overview), module constraints tied to requirements.

4 Testing

Testing is an **extremely** important component of most projects, whether it involves a circuit, a process, or software.

1. Define the needed types of tests (unit testing for modules, integrity testing for interfaces, user-study or acceptance testing for functional and non-functional requirements).
2. Define/identify the individual items/units and interfaces to be tested.
3. Define, design, and develop the actual test cases.
4. Determine the anticipated test results for each test case
5. Perform the actual tests.
6. Evaluate the actual test results.
7. Make the necessary changes to the product being tested
8. Perform any necessary retesting
9. Document the entire testing process and its results

Include Functional and Non-Functional Testing, Modeling and Simulations, challenges you have determined.

4.1 Unit Testing

– Discuss any hardware/software units being tested in isolation

4.2 Interface Testing

– Discuss how the composition of two or more units (interfaces) are to be tested. Enumerate all the relevant interfaces in your design.

4.3 Acceptance Testing

How will you demonstrate that the design requirements, both functional and non-functional are being met? How would you involve your client in the acceptance testing?

4.4 Results

– List and explain any and all results obtained so far during the testing phase

- Include failures and successes
- Explain what you learned and how you are planning to change the design iteratively as you progress with your project
- If you are including figures, please include captions and cite it in the text

5 Implementation

Describe any (preliminary) implementation plan for the next semester for your proposed design in 3.3.

6 Closing Material

6.1 Conclusion

Summarize the work you have done so far. Briefly re-iterate your goals. Then, re-iterate the best plan of action (or solution) to achieving your goals and indicate why this surpasses all other possible solutions tested.

6.2 References

List technical references and related work / market survey references. Do professional citation style (ex. IEEE).

6.3 Appendices

Any additional information that would be helpful to the evaluation of your design document.

If you have any large graphs, tables, or similar data that does not directly pertain to the problem but helps support it, include it here. This would also be a good area to include hardware/software manuals used. May include CAD files, circuit schematics, layout etc,. PCB testing issues etc., Software bugs etc.