Exposé for a Master's Thesis

Working Title:

# “Enhancing Automatic Bug Fixing by Integrating Spectral Analysis and Reinforcement Learning”

Aayush Manoj Tirmalle

December 16, 2024

## 1. Domain

The project is situated in the domain of Automated Software Engineering, with a specific focus on Automated Code Defect Identification and Fixing. It integrates Spectral Analysis[5], and Reinforcement Learning[4] techniques to enhance Large Language Models (LLMs) in program defect detection and automated repair.

## 2. Problem Description

Debugging and fixing software defects is one of the most time-consuming tasks in software development, often requiring significant manual effort and expertise. Existing Automated Program Repair (APR) systems leverage Large Language Models (LLMs) such as GPT-4, but their performance can be limited by inefficiencies in two critical areas:

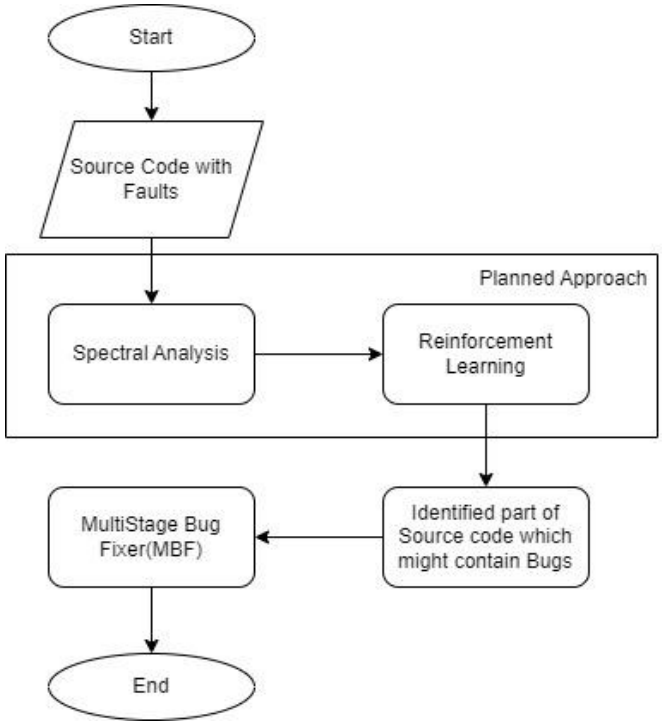
- Bug Localization: Many systems fail to pinpoint the exact sections of code responsible for errors. This results in LLMs being tasked with analyzing entire codebases, leading to reduced efficiency and suboptimal fixes.

- Context Understanding: LLMs struggle to handle long sequences of code effectively, which impairs their ability to provide accurate and robust bug fixes.

## 3. Goals

a. Achieve Accurate Bug Detection: Aim for precise localization of faults in source code to enhance defect identification accuracy, thereby reducing ambiguity in bug detection processes.
b. Enable Effective Bug Resolution: Ensure the development of robust, maintainable fixes for detected defects, thereby improving the reliability and correctness of the repair process.
c. Optimize Resource-Efficient Processing: Minimize computational costs by using less time and fewer tokens, enhancing the overall efficiency of the defect detection and fixing system.

## 4. Planned Approach




## 5. Work Packages and duration

a. **WP1**: Literature Review and Requirement Analysis (3 weeks)

*Objective*: To establish a solid understanding of existing methodologies and identify the requirements for the proposed framework.

*Tasks* :

- Conduct an in-depth literature review on automated program repair, LLM-based bug detection, and reinforcement learning in AI.

- Analyze benchmarks like QuixBugs to select suitable datasets for testing the framework's performance.
- Identify gaps in current methodologies and formulate research questions for the thesis.

b. **WP2**: Model Fine-Tuning and Development of Spectral Analysis Module (4 weeks)

Objective: To deploy a locally deployable LLM and Achieve accurate and efficient identification of buggy code segments.
Tasks:

- Select and fine-tune an open-source LLM (e.g., LLaMA) for bug detection and repair in specific programming languages and contexts.

- Implement fault localization techniques using spectral analysis and static data flow analysis to preprocess code and identify potential defect locations.

c. **WP3**: Developing Reinforcement Learning-Based Fault Localization (4 weeks)
Objective: Enhance the accuracy and reliability of fault detection through iterative learning.
Tasks:

- Design, develop and train an RL model to refine the output of the spectral analysis module.

- Integrate the RL agent with the spectral analysis module to create a unified fault localization pipeline.

d. **WP4**: Integration of LLM and Multi-Stage Bug Fixer[1] (4 weeks)

Objective: Seamlessly connect the fault localization module with the repair framework to ensure end-to-end functionality.
Tasks:

- Set up a Fine-tuned locally deployed Llama model to accept buggy code snippets localized by the RL agent.

- Integrate the Multi-Stage Bug Fixer (MBF) to handle the actual bug fixing, leveraging the preprocessed input for optimal performance.

e. **WP5**: Evaluation and Optimization (4 weeks)

Objective: Validate the system's robustness and identify areas for improvement.

Tasks:

- Test the complete pipeline on diverse datasets
- Compare the results

## 6. Literature

[1] Weng, G., & Andrzejak, A. Automatic Bug Fixing via Deliberate Problem Solving with Large Language Models. Heidelberg University, SAP SE, 2023.
[2] Yao, S., et al. Tree of Thoughts: Deliberate Problem Solving with Large Language Models. arXiv preprint, 2023.
[3] https://arxiv.org/pdf/2401.03374
[4] Reinforcement Learning for Mutation Operator Selection in Automated Program Repair Carol Hanna, Aymeric Blot, Justyna Petke
[5] Improving Spectral-Based Fault Localization Using Static Analysis Neelofar, Lee Naish, Jason Lee and Kotagiri Ramamohanarao