# A Three-Stage Genetic Algorithm for Compiler Flag and Library Version Selection to Minimize Execution Time

Chi Ho Chan[1, 2] and Spyro Nita[2]

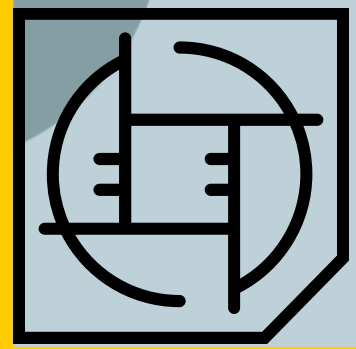
[1] Edinburgh Napier University, UK

[2] EPCC, The University of Edinburgh, UK

GI@ICSE 2025

Edinburgh Napier UNIVERSITY

Centre for Artificial Intelligence & Robotics



# Outline

- Motivations
- Proposed Algorithm
- Case Study
- Challenges & Future Work

# Outline

- **Motivations**
- Proposed Algorithm
- Case Study
- Challenges & Future Work

# Motivations

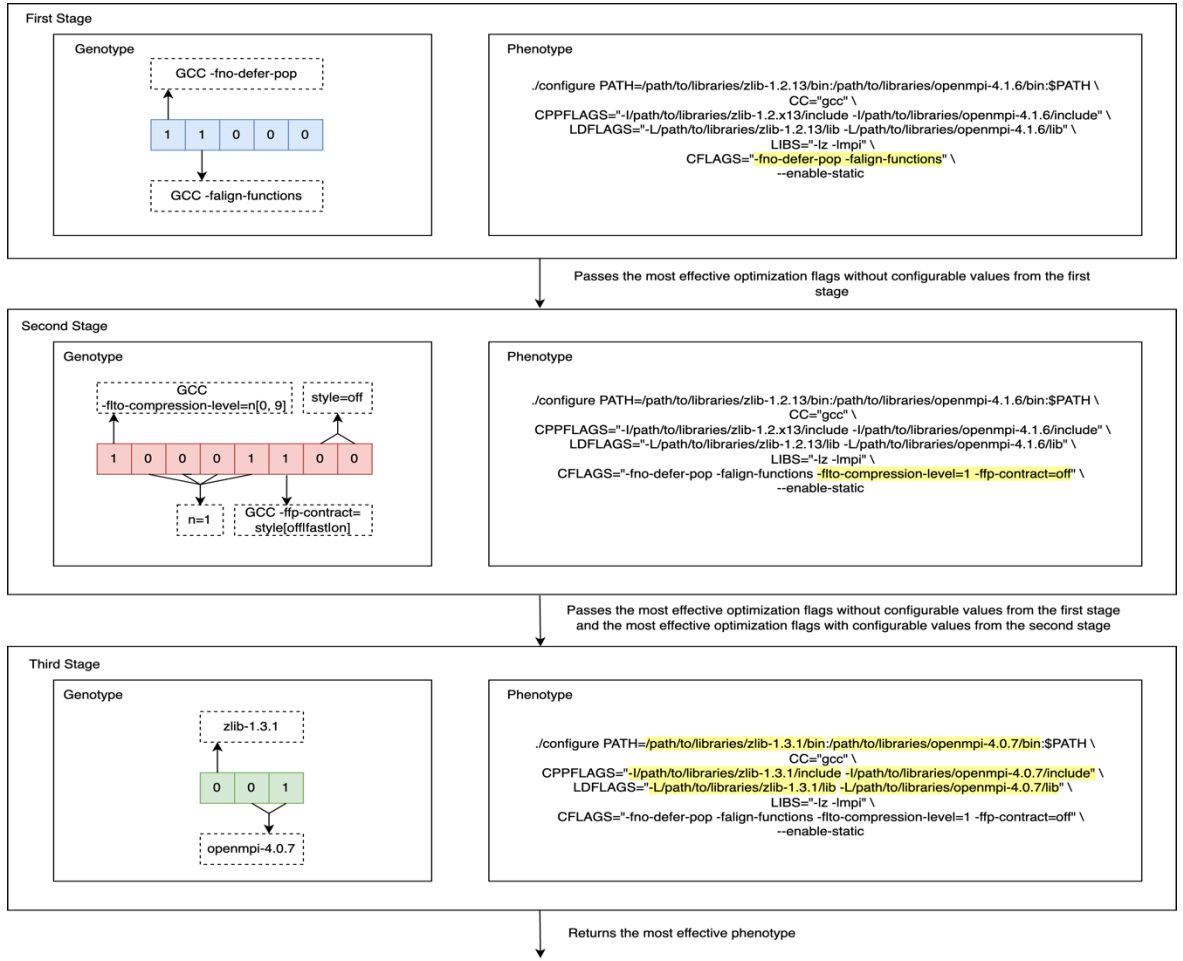
- Manual flag selection is complex and time-consuming.
- Optimization levels such as `-O1`, `-O2`, and `-O3` may not yield the optimal performance across all applications.
- Existing compiler autotuning methods mainly focus on selecting **optimization flags without configurable values** (e.g. `-fforward-propagate`).
- The optimization of the following flags are underexplored:
  - **Optimization flags with configurable values** (e.g. `-faline-functions=n`, `-faline-functions=n:m`)
  - **Link flags** (e.g. `-l`) and **directory flags** (e.g. `-I`, `-L`) for selecting library version

# Outline

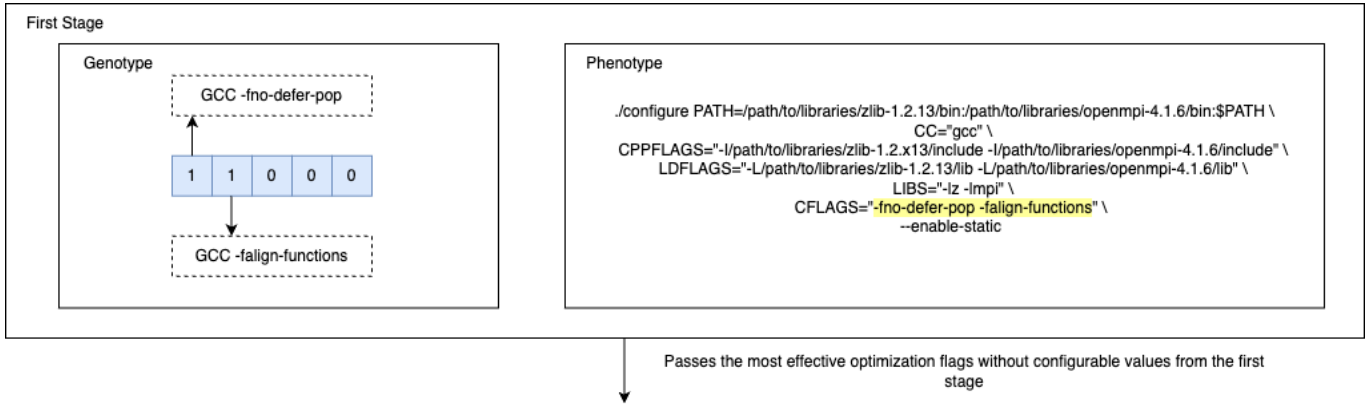
- Motivations
- **Proposed Algorithm**
- Case Study
- Challenges & Future Work

# Overview

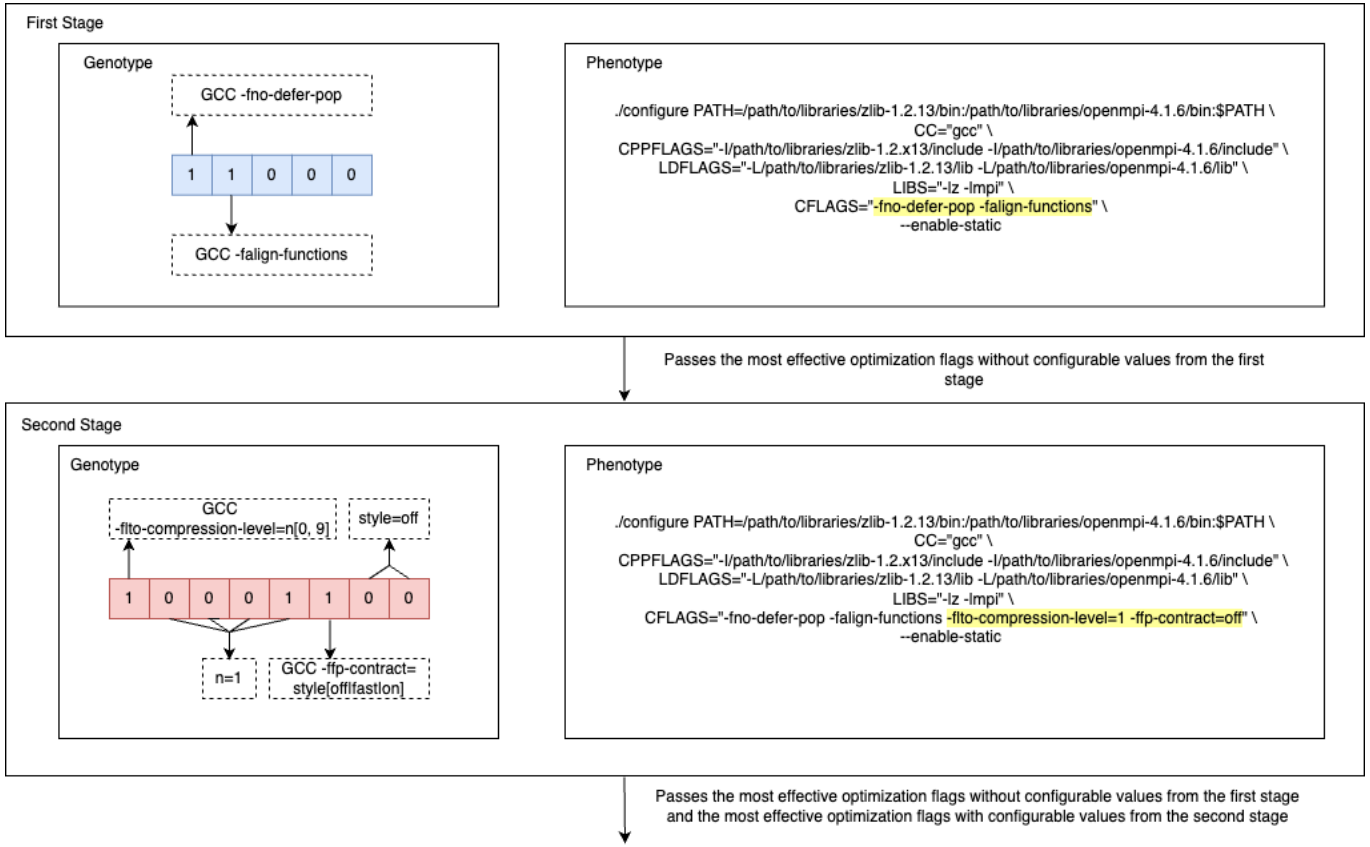
- **Genotype:** Binary string
- **Phenotype:** Command to run the configuration script
- **Fitness Function:** Returns average execution time across runs, or zero if any compilation or execution fails



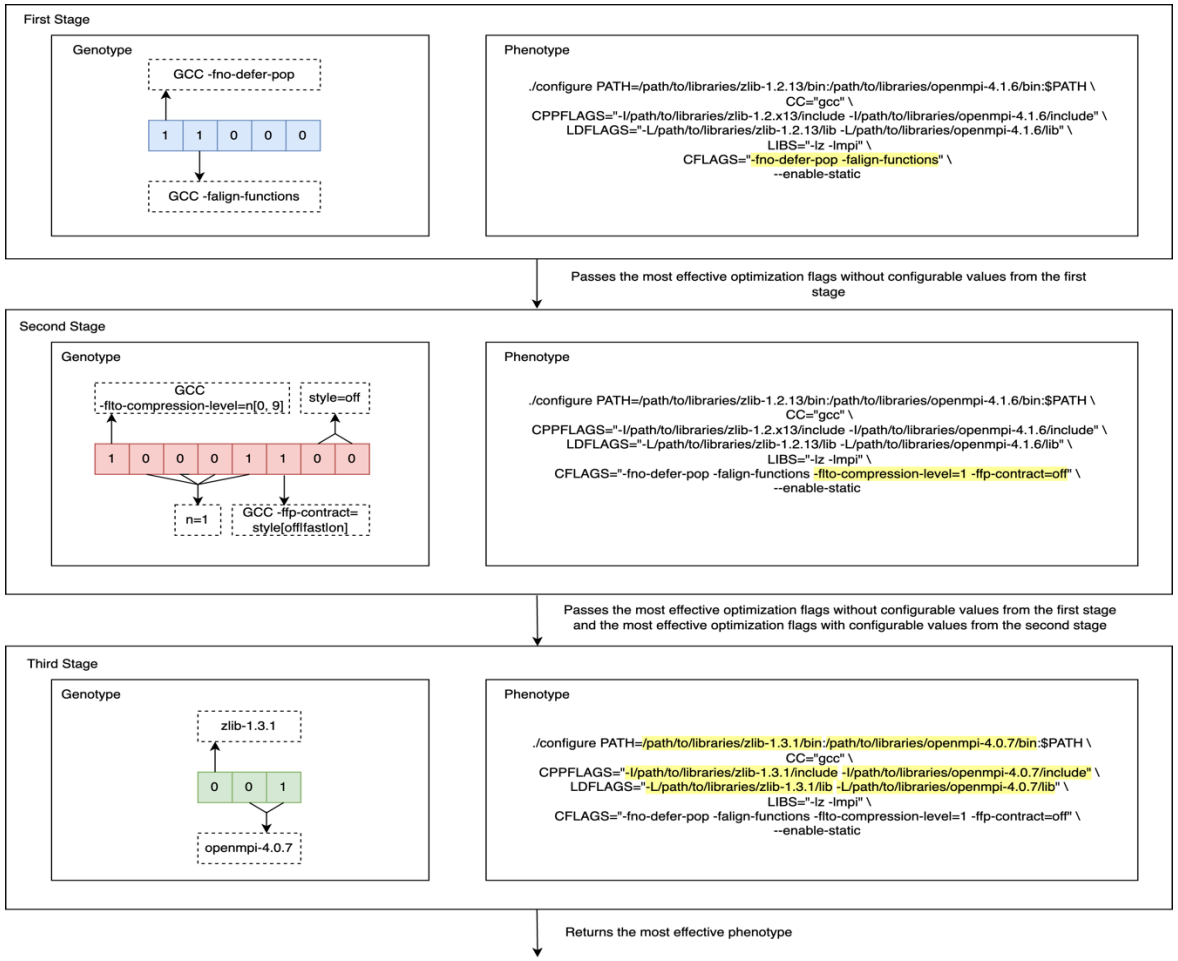
# Overview (Cont.)



# Overview (Cont.)

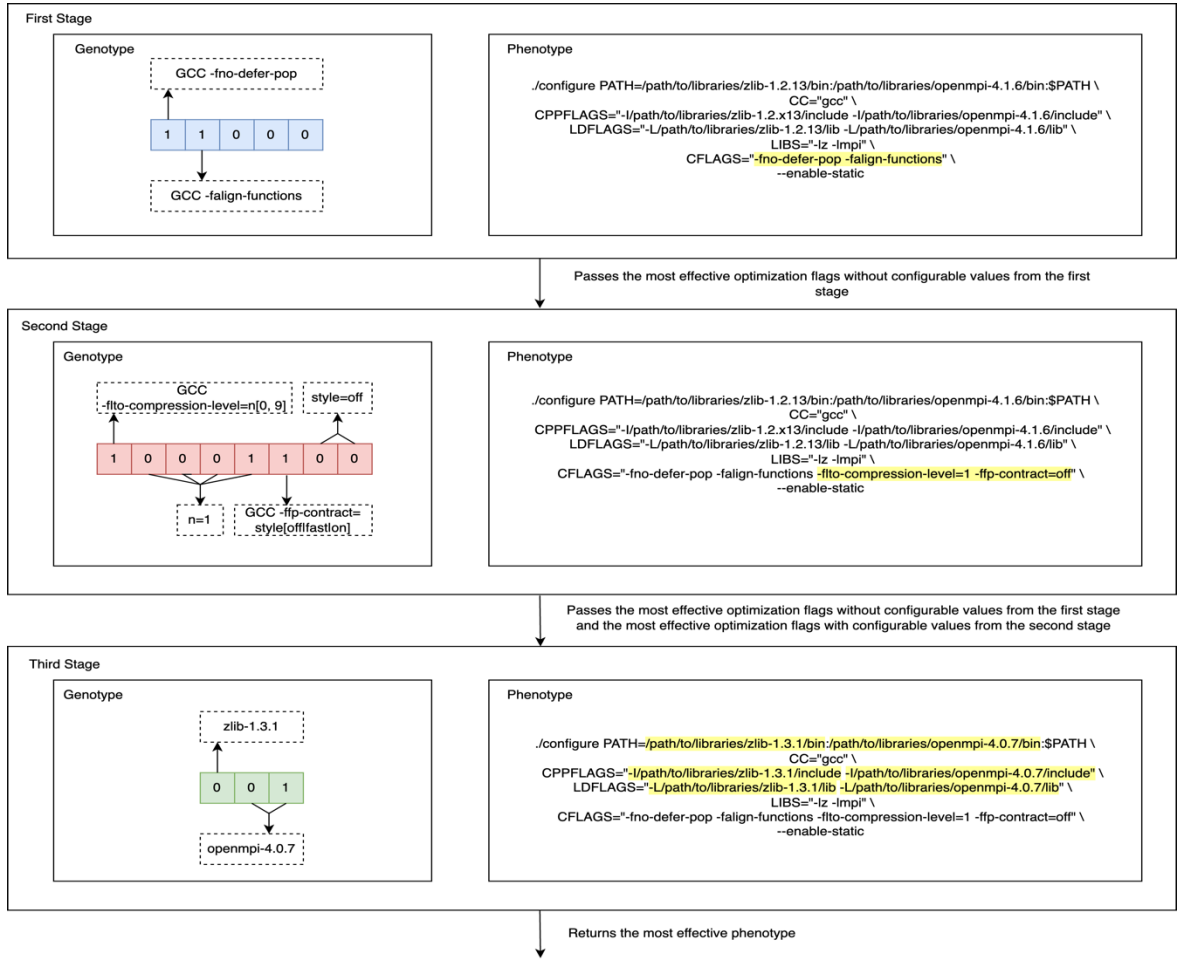


# Overview (Cont.)



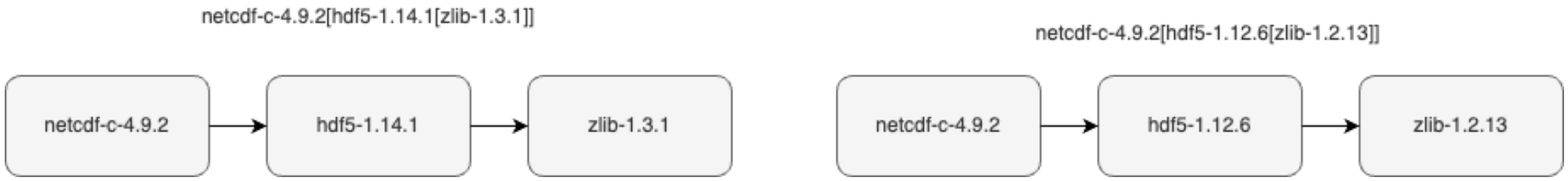
# Overview (Cont.)

- **Benefits:**
  - Reduces search space
  - Helps identify compilation and execution error
  - Incremental refinement



# Preprocessing Tasks

- **Optimization flag collection:**
  - Filters non-configurable flags that cause errors.
  - The final collection consists of filtered non-configurable flags and all configurable flags.
- **Library compilation:**
  - A unique directory name is required for each library version.
  - Each directory contains subdirectories, including `lib`, `include`, and `bin`.
  - Libraries are compiled and linked statically.



# Handling Configurable Flag

- Suppose there are *N* distinct values, where *N*!=2^*B* and *B* is the number of bits required. How does the genotype-to-phenotype mapping work?
- **Solution for *N*=6 and *B*=3:**

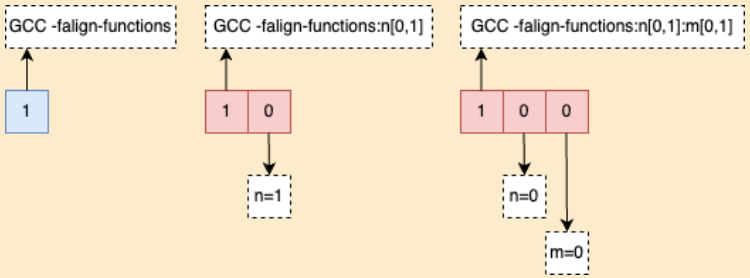
| Genotype | Phenotype |
|---|---|
| 000 | 0 |
| 001 | 1 |
| 010 | 2 |
| 011 | 3 |
| 100 | 4 |
| 101 | 5 |
| 110 | 0 |
| 111 | 1 |

# Handling Configurable Flag (Cont.)

- How should values ranging from 0 to infinity be handled?
- **Solution:** Infinity is replaced with a finite positive number.

# Handling Configurable Flag (Cont.)

- How should optimization flags with varying numbers of configurable values be handled?
- **Solution:**

| Genotype | Phenotype |
| --- | --- |
|  | ~~-faline-functions -faline-functions=1~~ -faline-functions=0:0 |

# Outline

- Motivations
- Proposed Algorithm
- **Case Study**
- Challenges & Future Work

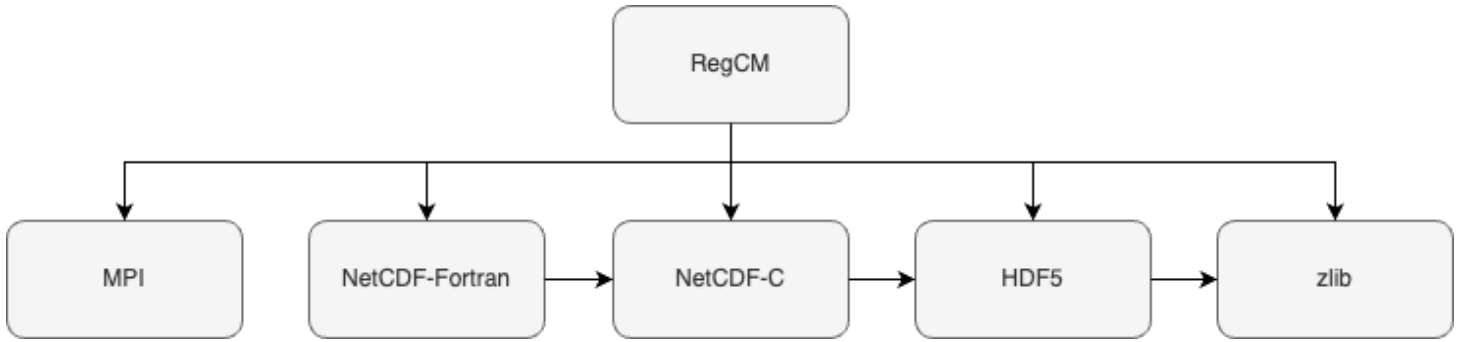
# One-Stage Genetic Algorithms (GAs)

- **GA-I:** Selects optimization flags without configurable value
- **GA-II:** Selects optimization flags with and without configurable value
- **GA-III:** Selects optimization flags with and without configurable value, and library versions.

| Hyperparamter | Value |
|---|---|
| GA type | Generational |
| Population size | 30 |
| Number of generations | 3 |
| Crossover rate | 0.9 |
| Mutation rate | 0.1 |
| Selection type | Tournament |
| Tournament size | 3 |
| Elitism count | 2 |
| Crossover type | Uniform |
| Mutation type | Random |
| Number of compilations in a fitness function call | 1 |
| Number of executions in a fitness function call | 3 |
| Upper bound used to replace infinity in the value range (For GA-II and GA-III) | 15 |

# Test Program: RegCM

- Simulates long-term regional climate
- Mainly written in Fortran, with some components in C
- **Dependent libraries:** NetCDF-C, NetCDF-Fortran, zlib, HDF5, and an MPI implementation
- **Main input parameters:** `X`, `Y`, and `Z` dimensions
- **Input parameter settings:** `jx=100` (`X` dimension), `iy=50` (`Y` dimension), `kz=30` (`Z` dimension), and more.



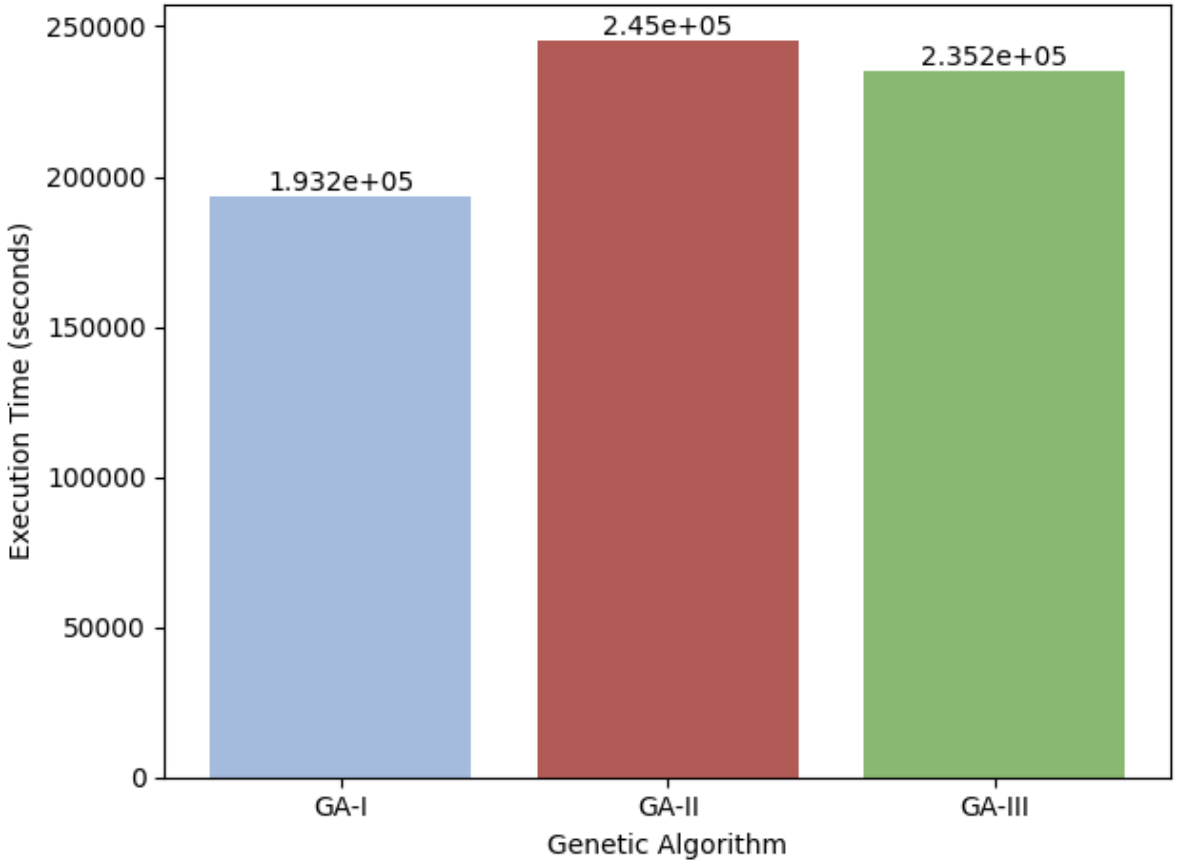
# Search Space

| GA | Number of GCC optimization flags without configurable value | Number of GFortran optimization flags without configurable value | Number of GCC optimization flags with configurable value | Number of GFortran optimization flags with configurable value | Number of library versions | Bits required for genotype |
|---|---|---|---|---|---|---|
| GA-I | 185 | 188 | 0 | 0 | 0 | 373 |
| GA-II | 185 | 188 | 19 | 8 | 0 | 606 |
| GA-III | 185 | 188 | 19 | 8 | 5 | 609 |

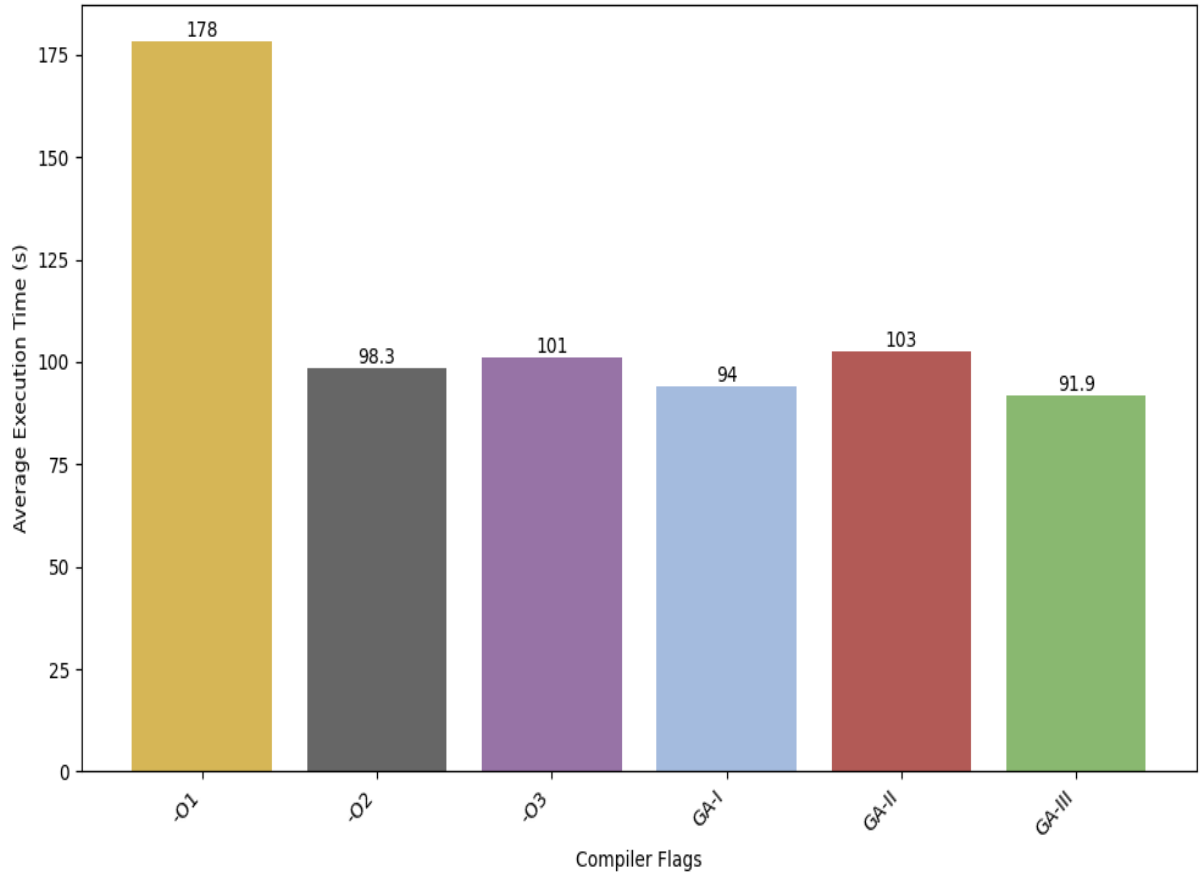
# Benchmarking Process

- **Processors:** 2x Intel Xeon Broadwell, 2.1 GHz, 18-core.
- Execute each GA once.
- Compare six configurations: flags generated by GA-I, GA-II, GA-III, and the optimization levels `-O1`, `-O2`, and `-O3`.
- For each configuration, compile RegCM with that configuration and repeat the execution ten times on 32 processes.

# Results – GA Runtime



# Results – RegCM Runtime



# Outline

- Motivations
- Proposed Algorithm
- Case Study
- **Challenges & Future Work**

# Challenges & Future Work

- **Challenges:**
  - Handling infinite ranges efficiently.
  - Ensuring flag reliability.
- **Future Work:**
  - Empirical studies
  - Alternative optimization objectives
  - Variable-length genotypes / specialized genotype-to-phenotype mapping
  - Additional factors such as command-line interface (CLI) options preprocessing directives, and environment variables

Edinburgh Napier UNIVERSITY | Centre for Artificial Intelligence & Robotics

# Thank you!

www.napier.ac.uk/CentreENU