

HANOI UNIVERSITY OF SCIENCE AND TECHNOLOGY

GRADUATION THESIS

Evolutionary Computation and Large Language Model for Automatic Heuristics Design

PHẠM VŨ TUẤN ĐẠT
dat.pvt210158@sis.hust.edu.vn

Major: Computer Science

Supervisor: Associate Professor Huỳnh Thị Thanh Bình

Signature

Department: Computer Science

School: School of Information and Communications Technology

HANOI, 01/2025

ACKNOWLEDGMENT

For me, finishing this project signifies not just a milestone but the start of an exciting new journey. While conducting this thesis, I have often felt lucky. This work would not have been possible without the invaluable guidance and mentorship of Associate Professor Huỳnh Thị Thanh Bình. I am also grateful to my mentor, Long Đoàn, for our weekly discussions and his thoughtful feedback. I would like to extend my thanks to the members of the MSO Lab, who offered their time to enrich this work in countless ways. Last but not least, I want to express my heartfelt appreciation to my beloved family and best friends, who have always been my source of emotional support and encouragement during the most challenging times. **Thank you all from the bottom of my heart!**

ABSTRACT

Automatic Heuristic Design (AHD) is an active research area due to its utility in solving complex search and NP-hard combinatorial optimization problems in the real world. Recent advances in Large Language Models (LLMs) introduce new possibilities by coupling LLMs with Evolutionary Computation (EC) to automatically generate heuristics, known as LLM-based Evolutionary Program Search (LLM-EPS). While previous LLM-EPS studies obtained great performance on various tasks, there is still a gap in understanding the properties of heuristic search spaces and achieving a balance between exploration and exploitation, which is a critical factor in large heuristic search spaces. This research addresses this gap by proposing two diversity measurement metrics and performing an analysis of previous LLM-EPS approaches, including FunSearch, EoH, and ReEvo. Results on black-box AHD problems reveal that while EoH demonstrates higher diversity than FunSearch and ReEvo, its objective score is unstable. Conversely, ReEvo's reflection mechanism yields good objective scores but fails to optimize diversity effectively. In light of these findings, HSEvo was introduced as an adaptive LLM-EPS framework that strikes a balance between diversity and convergence through the use of a harmony search algorithm. Experimentation demonstrated that HSEvo achieved high diversity indices and good objective scores while remaining cost-effective. These results underscore the importance of balancing exploration and exploitation and understanding heuristic search spaces in designing frameworks in LLM-EPS.

Student

Phạm Vũ Tuấn Đạt

TABLE OF CONTENTS

INTRODUCTION.....	1
CHAPTER 1. LITERATURE REVIEW	6
1.1 Related works.....	6
1.2 Evolutionary computation	7
1.2.1 Genetic algorithm	8
1.2.2 Evolutionary programming	9
1.2.3 Genetic programming.....	10
1.3 Large language models.....	10
1.4 LLM-based evolutionary program search	13
1.4.1 FunSearch	13
1.4.2 Evolution of heuristics.....	15
1.4.3 Reflective evolution	16
1.5 Diversity in evolutionary computation	17
CHAPTER 2. DIVERSITY MEASUREMENT METRICS.....	19
2.1 Population encoding	19
2.2 Shannon–Wiener diversity index.....	19
2.3 Cumulative diversity index	20
2.4 Exploring the correlation between objective score and diversity measure- ment metrics.....	22
2.5 Diversity analysis on previous LLM-EPS framework	23
CHAPTER 3. AUTOMATIC HEURISTIC DESIGN WITH HSEVO.....	26
3.1 Automatic heuristic design with HSEvo.....	26
3.2 HSEvo prompt examples	32
3.2.1 Task description prompts	32

3.2.2 Population initialization prompt.....	33
3.2.3 Flash reflection prompts	33
3.2.4 Crossover prompt.....	35
3.2.5 Elitist mutation prompt.....	36
3.2.6 Harmony search prompts	36
3.2.7 Problem-specific prompts	37
3.3 Generated heuristics.....	38
CHAPTER 4. NUMERICAL RESULTS.....	48
4.1 Experimental settings.....	48
4.2 Benchmark dataset details	48
4.2.1 Bin packing online	48
4.2.2 Traveling salesman problem.....	49
4.2.3 Orienteering problem	50
4.3 Experimental results	51
4.4 Ablation study	52
4.4.1 Harmony search analysis.....	52
4.4.2 Flash reflection analysis	53
CONCLUSIONS	55
REFERENCE	60

LIST OF FIGURES

Figure 1.1	Flow chart of GA.	8
Figure 1.2	Architecture of large neural models	12
Figure 1.3	An illustration of the LLM-based EPS paradigm, with re- spect to the GP-based paradigm (top section), for AHD.	13
Figure 1.4	The FunSearch process	14
Figure 1.5	Program clusters within islands of FunSearch	14
Figure 1.6	An illustration of the AEL framework	15
Figure 1.7	EoH evolves both thoughts and codes using LLMs.	15
Figure 1.8	ReEvo pipeline	16
Figure 2.1	Diversity indices and objective scores of ReEvo framework on BPO problem through different runs.	23
Figure 2.2	Diversity indices and objective scores of ReEvo framework on TSP through different runs.	23
Figure 2.3	Diversity indices and objective scores of ReEvo framework on OP through different runs.	24
Figure 2.4	CDI and objective scores of previous LLM-EPS on different AHD problems.	24
Figure 3.1	Overview of the HSEvo framework.	27
Figure 3.2	An example of how the harmony search component works in HSEvo.	39
Figure 3.3	Function signatures used in HSEvo for BPO.	39
Figure 3.4	Function signatures used in HSEvo for TSP.	39
Figure 3.5	Function signatures used in HSEvo for OP.	40
Figure 3.6	Seed heuristics used in HSEvo for BPO.	40
Figure 3.7	Seed heuristics used in HSEvo for TSP.	41
Figure 3.8	Seed heuristics used in HSEvo for OP.	41
Figure 3.9	The best heuristic for BPO found by HSEvo.	42
Figure 3.10	The best heuristic for BPO found by HSEvo (continue). . . .	43
Figure 3.11	The best heuristic for TSP found by HSEvo.	44
Figure 3.12	The best heuristic for TSP found by HSEvo (continue). . . .	45
Figure 3.13	The best heuristic for OP found by HSEvo.	46
Figure 3.14	The best heuristic for OP found by HSEvo (continue).	47

Figure 4.1	Diversity indices and objective scores of HSEvo framework on BPO problem through different runs.	51
Figure 4.2	Diversity indices and objective scores of HSEvo framework on TSP problem through different runs.	51
Figure 4.3	Diversity indices and objective scores of HSEvo framework on OP problem through different runs.	52

LIST OF TABLES

Table 3.1	Problem descriptions used in prompts	37
Table 3.2	Function descriptions used in prompts	37
Table 4.1	Summary of parameters settings	49
Table 4.2	Problems parameters used for heuristic evaluations	50
Table 4.3	Experiment results on different AHD problems	51
Table 4.4	Ablation results on harmony search	52
Table 4.5	Ablation results on flash reflection	53

LIST OF ABBREVIATIONS

Abbreviation	Definition
ACO	Ant Colony Optimization
AHD	Automatic Heuristic Design
AI	Artificial Intelligence
BPO	Bin Packing Online
CDI	Cumulative Diversity Index
COPs	Combinatorial Optimization Problems
EC	Evolutionary Computation
EP	Evolutionary Programming
ES	Evolution Strategies
GA	Genetic Algorithm
GLS	Guided Local Search
GP	Genetic Programming
HHs	Hyper-Heuristics
HS	Harmony Search
LLM-EPS	LLM-based Evolutionary Program Search
LLMs	Large Language Models
MST	Minimum Spanning Tree
NCO	Neural Combinatorial Optimization
OP	Orienteering Problem
SDI	Shannon Diversity Index
SWDI	Shannon-Wiener Diversity Index
TSP	Traveling Salesman Problem

INTRODUCTION

NP-hard Combinatorial Optimization Problems (COPs) frequently arise in real-world scenarios. Examples include groups of problems such as routing and logistics, scheduling and planning, network design, and others. To address these problems, heuristic methods are among the most commonly employed approaches due to their superior efficiency in terms of time and resources needed to find solutions that approximate the optimal. Over the past few decades, significant efforts have been dedicated to designing effective search methods, resulting in techniques such as simulated annealing, tabu search, and iterated local search, among many others. These manually crafted methods have been successfully applied in numerous real-world applications.

However, due to the diverse nature of practical problems, each application comes with its constraints and objectives, often requiring customization or the selection of specific search methods tailored to the situation at hand. Manually creating, tuning, and configuring search methods for particular problems is not only labor-intensive but also demands deep expertise. This represents a bottleneck in many application domains. Consequently, AHD has emerged as a promising solution, aiming to automate the selection, tuning, or construction of efficient search methods for specific classes of problems.

The goal of AHD is to address the difficulties involved in the creation of manual heuristics by utilizing computational methods to automatically generate or enhance heuristics. Among the various approaches within AHD, both Hyper-Heuristics (HHs) [1], [2] and Neural Combinatorial Optimization (NCO) [3]–[5] have gained substantial attention for their potential to address the limitations of traditional heuristic development.

HHs operate within predefined heuristic spaces curated by human experts, selecting or generating heuristics from these established sets. While HHs have demonstrated success, their performance is inherently limited by the scope and quality of the predefined heuristic space, often restricting the discovery of novel or more effective heuristics.

On the other hand, NCO employs neural networks to learn patterns and predict solutions for COPs. By leveraging data-driven models, NCO seeks to generalize across problem instances, offering a more dynamic approach to heuristic development. Despite its promise, NCO faces significant challenges. Effective generalization across diverse problem instances requires robust inductive biases [6], which