Vellore Institute of Technology, Chennai
School of Computing Science and Engineering

Keyboard Layout Optimization
Using Genetic Algorithms and Particle Swarm Optimization

Submitted by
Ashish (22BAI1062)
Ayush Bist (22BAI1275)

Under the Supervision of
Dr. Rajalakshmi R
Professor, VIT Chennai

November 2024

Abstract

This research investigates the optimization of keyboard layouts through Genetic Algorithms (GA) and Particle Swarm Optimization (PSO) to improve typing efficiency on staggered and ortho-linear keyboard configurations. Efficiency is quantified by minimizing the distance traveled by fingers across a dataset of 100,000 research paper abstracts sourced from various academic fields. Two primary objectives guide the layout optimization process: prioritizing efficiency alone and balancing efficiency with layout similarity to the conventional QWERTY arrangement. Results indicate notable efficiency gains, particularly for ortho-linear layouts optimized without similarity constraints. Additionally, PSO struggles to converge under similarity constraints, diverging instead. Overall, GA emerges as the more stable and effective method, demonstrating robust performance in both similarity-constrained and unconstrained scenarios.

Introduction

Traditional keyboard layouts, most notably the QWERTY design, were developed without prioritizing typing efficiency, which has driven a growing body of research aimed at optimizing keyboard layouts. Despite its inefficiencies, QWERTY remains the standard largely due to familiarity rather than performance. Numerous studies highlight the need for more ergonomic designs that reduce finger movement, thereby enhancing typing speed and comfort.

Optimization techniques such as Genetic Algorithms (GA) and Particle Swarm Optimization (PSO) have been applied to the keyboard layout problem to generate layouts that minimize finger travel distance while potentially maintaining a familiar layout structure. This study focuses on optimizing layouts for two physical configurations—staggered and ortho-linear—using GA and PSO algorithms under efficiency-only and similarity-constrained conditions. The final results are evaluated for finger travel efficiency, similarity to QWERTY, and layout stability.

Literature Review

A variety of keyboard layouts, including DVORAK and alphabetical layouts, have been developed to improve typing efficiency. According to Buzing (2003), DVORAK offers minimal speed advantages over QWERTY for expert typists, while alphabetical layouts assist novice typists in locating keys more intuitively. This research underscores the complexity of developing a universally efficient layout that balances typing speed with user familiarity.

Ergonomics-driven designs, such as split keyboards, have been successful due to their health benefits and reduced strain, as discussed by Rempel (2008). This line of ergonomic research provides insights into the value of reducing strain and improving comfort through thoughtful layout design.

Optimization techniques such as GA and PSO have shown considerable promise in layout and configuration tasks. Moradi and Nickabadi (2006) demonstrated that GAs could effectively optimize mobile keypad layouts for faster typing. Onsorodi and Korhan (2020) further showed that GAs could enhance keyboard ergonomics by reducing finger travel distance compared to QWERTY. Similarly, Gad (2022) reviewed PSO applications, highlighting its broad utility in optimization tasks, though it can sometimes diverge when tasked with multiple, potentially conflicting objectives.

Bi, Smith, and Zhai (2010) explored "Quasi-QWERTY" optimization to improve soft keyboard layouts by minimizing the learning curve for users, and Zhai, Hunter, and Smith (2002) demonstrated the impact of advanced virtual keyboard design techniques on efficiency. Together, these studies emphasize the balance between optimizing for efficiency and maintaining layout familiarity for users.

Methodology

Keyboard Layout Representation

Each keyboard layout is represented as a string of 26 letters corresponding to a left-to-right order across three rows (top, home, bottom), allowing each layout to function as an individual in the GA or as a particle in PSO. Default or QWERTY is represented as “qwertyuiopasdfghjklzxcvbnm”.

Physical Layouts

1. Staggered Layout: This configuration mimics the standard QWERTY arrangement, where rows are horizontally offset.
2. Ortho-Linear Layout: All keys are aligned vertically, reducing the lateral movement required during typing.

Fitness Function

Efficiency is measured by calculating the total finger travel distance over a dataset of 100,000 research paper abstracts across ten fields. Two fitness functions guide the optimization:
1. Efficiency-Only (NoSim): Solely minimizes travel distance.
2. Similarity-Constrained (Sim): Minimizes travel distance while maintaining a degree of similarity to QWERTY, quantified by a similarity score (1.0 indicates full similarity).

Optimization Algorithms

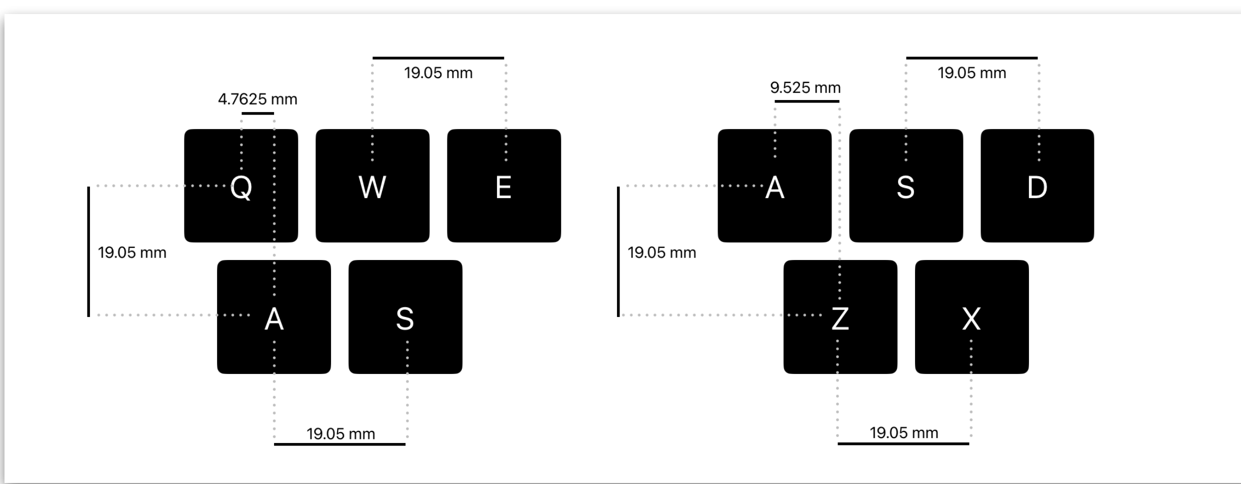
1. Genetic Algorithm (GA):
GA operates with a population size of 50, running for 200 generations. Each layout undergoes mutation and crossover operations, with the fitness calculated based on finger travel and similarity requirements as applicable.

2. Particle Swarm Optimization (PSO):
PSO uses a swarm of 50 particles for up to 200 iterations, adjusting layout positions based on personal and global best solutions. Under similarity constraints, PSO exhibited frequent divergence, particularly in the ortho-linear layout, indicating potential limitations in handling multi-objective optimization for this task.

Evaluation Metrics

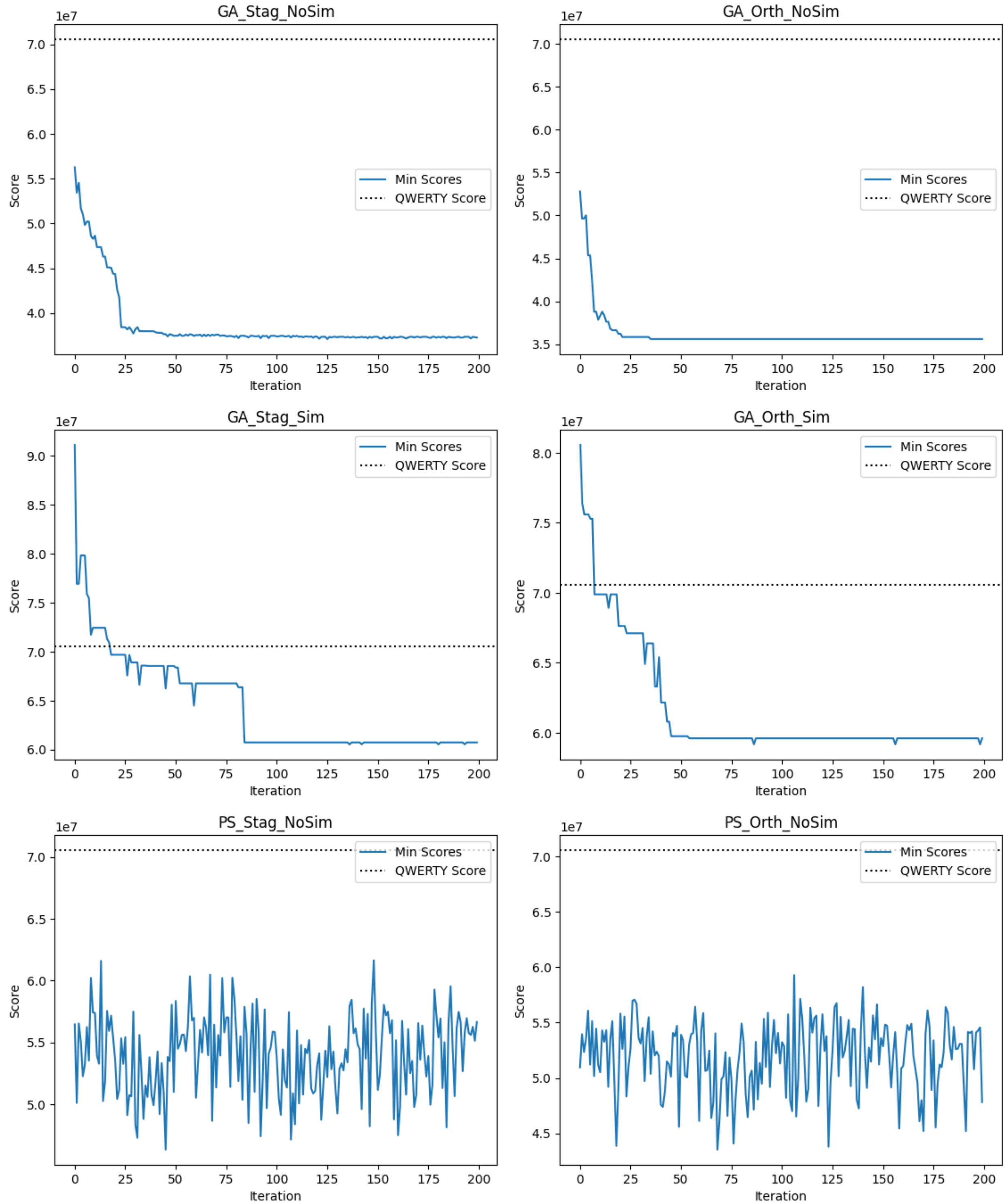
Efficiency is measured in units based on the 1u key size (18.05 mm), with distance calculated from the center of one key to another. Results are compared against QWERTY on staggered layouts, allowing efficiency gains to be expressed as percentage reductions from the QWERTY baseline.

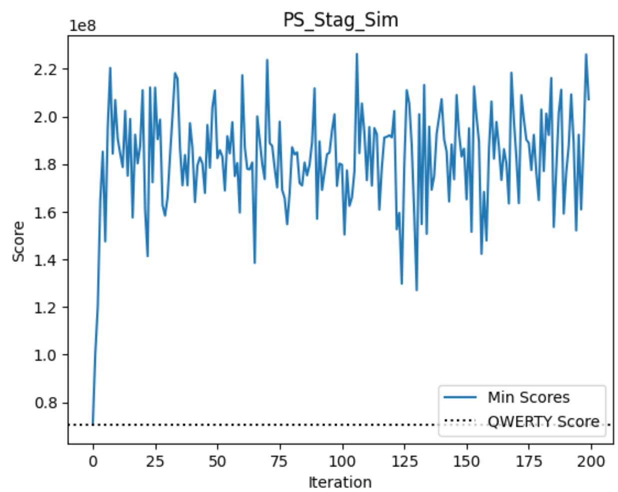
Figure 1. Distance between Key Centers



Results

Figure 2-9. Scores progression of different parameters.





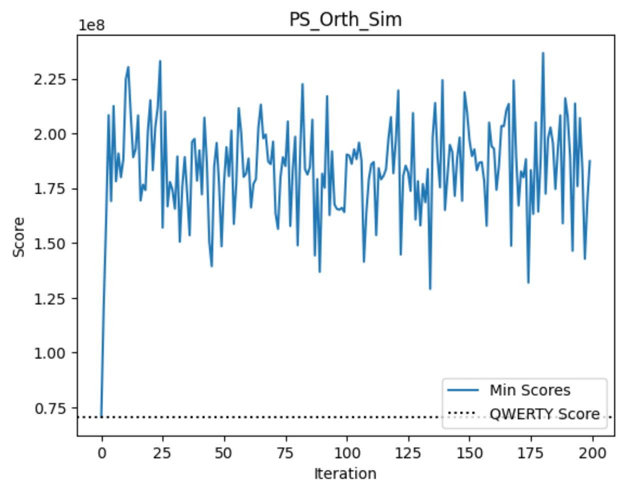


Table 1. Optimized Layout Results

SrNo	Algorithm	Layout	Score	Similarity
1	GA_Stag_NoSim	hwbukxmdgcsionrlaetpyvfjzq	36,982,817	0.192308
2	GA_Orth_NoSim	wphvzxdbgcstenmuiaorylfqjk	35,591,180	0.307692
3	GA_Stag_Sim	qwbryxuompeasdfgitnhjklzcv	56,455,814	0.730769
4	GA_Orth_Sim	qmcdwyuiopastrfghenjklzxvb	56,534,393	0.730769
5	PS_Stag_NoSim	vhfxwjkrlmseioaztyndgqpbcu	44,384,686	0.192308
6	PS_Orth_NoSim	yxwhkjqcaroetsdzpinlgmufbv	41,432,902	0.269231
7	PS_Stag_Sim	qwertyuiopasdfghjklzxcvbnm	70,562,168	1.000000
8	PS_Orth_Sim	qwertyuiopasdfghjklzxcvbnm	70,562,168	1.000000

Analysis

Efficiency Improvements

- Efficiency-Focused Layouts (NoSim): The ortho-linear layout optimized by GA (GA_Orth_NoSim) achieved the highest efficiency, reducing travel distance by 49.5% compared to QWERTY staggered. The GA staggered layout (GA_Stag_NoSim) also performed well, showing a 47.5% reduction.

- Similarity-Constrained Layouts (Sim): The GA ortho-linear layout achieved a 20% reduction from QWERTY while maintaining a similarity score of 0.73.

Algorithmic Stability

GA outperformed PSO in stability, particularly in similarity-constrained tasks. PSO's tendency to diverge under similarity constraints, particularly for ortho-linear layouts, highlights its limitations in complex, multi-objective optimization problems, aligning with observations in previous PSO research by Gad (2022).

Discussion

Both GA and PSO are effective for optimizing keyboard layouts, though their performance varies by layout type and objective. GA consistently delivered robust results across different layouts, while PSO struggled with convergence when similarity constraints were applied. This divergence likely arises from the dynamic adjustments based on personal and global best positions, which can introduce large positional shifts under similarity constraints.

Conclusion

The results confirm that significant efficiency improvements are achievable through optimized keyboard layouts, with GA and ortho-linear configurations yielding the most efficient solutions. GA's stable performance under all tested conditions, especially with similarity constraints, positions it as the preferred optimization algorithm for keyboard layouts. This study contributes to the field of keyboard optimization by validating the advantages of GA in multi-objective optimization scenarios and offering a comparative benchmark for future studies on layout efficiency.

Future work may explore hybrid algorithms that blend GA's exploration capabilities with PSO's adaptive dynamics for more comprehensive optimization.

References

- Buzing, P. (2003). "Comparing Different Keyboard Layouts: Aspects of QWERTY, DVORAK and alphabetical keyboards."

- Rempel, D. (2008). "The Split Keyboard: An Ergonomics Success Story." Human Factors, 50(3), 385-392.
- Bi, X., Smith, B. A., & Zhai, S. (2010). "Quasi-qwerty soft keyboard optimization." CHI '10.
- Gad, A.G. (2022). "Particle Swarm Optimization Algorithm and Its Applications: A Systematic Review." Arch Computat Methods Eng, 29, 2531–256
-
- 1.
- Moradi, S., & Nickabadi, A. (2006). "Optimization of Mobile Phone Keypad Layout via Genetic Algorithm."
- Onsorodi, A. H. H., & Korhan, O. (2020). "Application of a genetic algorithm to the keyboard layout problem."

Link

Project - Keyoard Layout Optimisation