Research Diary

PhD Research Journal

Author: **PhotonZhang**

Start Date: 29 July 2025

Field: **Industrial Engineering**

Contents

1 29 July 2025

1.1 Research Plan

Today's main tasks:

- Reading relevant literature on optimization methods
- Conducting experimental validation of production systems
- Summarizing research progress and planning next steps

1.2 Content Details

Paper Reading

Paper Title: Optimization Methods for Production Systems [1]
Authors: Smith, J. et al.
Main Content:

- Proposed a new optimization algorithm for production scheduling
- Achieved significant improvements in manufacturing efficiency
- Algorithm has better convergence and computational efficiency
- Introduced the concept of adaptive scheduling for dynamic environments

Personal Thoughts: The optimization approach in this paper shows promising results for industrial applications. The adaptive scheduling concept addresses dynamic production challenges effectively. This method might be applicable to my research on manufacturing systems.
Relevance Score: ***** (5/5)
Related Papers:

- [2] - Traditional scheduling methods comparison
- [3] - Baseline optimization approaches

Experiment Log

Experiment Name: Production System Optimization
Objective: Compare performance of different optimization algorithms for production scheduling

Experimental Setup:

- Dataset: Manufacturing production data (1000 orders, 50 machines)
- Algorithms: Genetic Algorithm [1], Simulated Annealing [2]
- Iterations: 1000 iterations
- Parameters: Adaptive mutation rate and cooling schedule
- Objective: Minimize makespan and maximize resource utilization

Results:

Algorithm	Makespan	Computational Time
Genetic Algorithm	92.3% efficiency	2.5h
Simulated Annealing	89.7% efficiency	3.2h

Key Findings:

- Genetic Algorithm performs significantly better due to adaptive parameters
- Computational time is reduced by 22% with improved convergence
- Better solution quality observed with genetic approach

Issues and Thoughts: Genetic Algorithm's superior performance validates the effectiveness of evolutionary optimization. Next step is to analyze the specific reasons and potentially apply hybrid methods [4] to further improve performance.

Code Snippet

```
% Example code for genetic algorithm implementation def
genetic_algorithm(population, generations=100):  """ Genetic algorithm
implementation for production scheduling """ best_solution = None
% Main evolution loop for generation in range(generations):  % Selection
parents = select_parents(population)
% Crossover offspring = crossover(parents)
% Mutation offspring = mutate(offspring)
% Evaluation fitness = evaluate(offspring)
% Update population population = update_population(population, offspring)
% Track best solution if fitness > best_fitness:  best_solution =
offspring best_fitness = fitness
return best_solution
```

Important Note

Important Discovery: The genetic algorithm approach shows promise for my research direction. Consider exploring:

- Hybrid optimization methods combining multiple algorithms
- Machine learning techniques [5] for parameter tuning
- Real-time scheduling applications

Daily Summary

Today's Achievements:

- Gained deep understanding of genetic algorithm principles from [1]
- Completed baseline experiments, establishing performance benchmarks
- Discovered promising research direction for hybrid optimization
- Identified potential applications in manufacturing systems

Tomorrow's Plan:

- Analyze specific reasons for Genetic Algorithm's performance advantages
- Implement hybrid optimization methods
- Prepare presentation materials for next week's group meeting
- Review related literature on optimization methods [6]

Issues Encountered: Some problems with experimental environment configuration, need to contact IT department for resolution. Computational resource limitations require optimization of algorithm parameters.
Inspiration Notes: Consider integrating machine learning techniques [4] into optimization algorithms, might yield unexpected results. The combination of evolutionary algorithms and machine learning could be a novel contribution to the field.

References

[1] Kaiming He et al. "Deep residual learning for image recognition". In: *Proceedings of the IEEE conference on computer vision and pattern recognition* (2016), pp. 770–778.

[2] Karen Simonyan and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition". In: *arXiv preprint arXiv:1409.1556* (2014).

[3] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. "Imagenet classification with deep convolutional neural networks". In: *Advances in neural information processing systems*. 2012, pp. 1097–1105.

[4] Ashish Vaswani et al. "Attention is all you need". In: *Advances in neural information processing systems* 30 (2017).

[5] Ian Goodfellow et al. "Generative adversarial nets". In: *Advances in neural information processing systems* 27 (2014).

[6] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. "Deep learning". In: *nature* 521.7553 (2015), pp. 436–444.