

University of Sheffield

# Optimized PI Controller Tuning for Electric Vehicle using NSGA-II



Mafaz Syed (230118234)

*Lecturer:* Prof Robin Purshouse

A report submitted in partial fulfillment of the requirements for the  
Decision Systems Assignment for ACS6124

*in the*

Automatic Control and Systems Engineering Department

May 20, 2024

## Executive Summary

This report presents the findings from a comprehensive tuning process of PI controller gains, aimed at enhancing the performance of an electric vehicle's control system. Utilizing multi-objective optimization, specifically the Non-Dominated Sorting Genetic Algorithm II (NSGA-II), PI controller settings that meet a range of performance criteria stipulated by the chief engineer were identified.

The primary goal was to optimize key performance indicators such as rise time, peak time, overshoot, undershoot, settling time, steady-state error, and control effort. The NSGA-II algorithm allowed us to explore and quantify the trade-offs inherent in simultaneously optimizing these competing objectives. The optimization results achieved the following:

- Majority of the performance criteria were met, with specific exceptions in rise time due to inherent trade-offs with control effort.
- The optimized PI controller settings,  $[K_P = 0.397, K_I = 0.253]$ , provide a balanced solution meeting the stringent requirements for stability and transient performance, albeit with slight compromises in the rise time.

For scenarios demanding greater sustainability, recommendations include adjusting the controller settings to  $[K_P = 0.000205, K_I = 0.12]$ , which meets the elevated sustainability targets but impacts the dynamic performance by increasing rise and peak times.

Given these findings, we recommend adopting the base PI controller settings for general applications to achieve an optimal balance between performance and energy efficiency. For enhanced sustainability goals, the alternative settings should be considered, keeping in mind the potential impact on transient performance.

Further real-time simulations and field tests are advised to refine these settings under various operational conditions, ensuring that the controller performance robustly supports the vehicle's operational needs. This study underscores the use of optimization tools for critical decision-making in electric vehicle design and highlights the potential for broader application across different aspects of vehicle systems engineering.

# Contents

<b>1</b>	<b>Multi-objective Optimisation for Engineering Design</b>	<b>1</b>
1.1	Comparison with Traditional Decision-Making Approaches . . . . .	1
1.2	Applications in Electric Vehicle Design . . . . .	1
1.3	Approaches to Multi-Objective Optimisation . . . . .	2
<b>2</b>	<b>Problem Formulation</b>	<b>3</b>
<b>3</b>	<b>Sampling Plan</b>	<b>4</b>
<b>4</b>	<b>Knowledge Discovery</b>	<b>6</b>
<b>5</b>	<b>Optimisation Process</b>	<b>8</b>
<b>6</b>	<b>Optimisation Results</b>	<b>10</b>
<b>7</b>	<b>Sustainability Analysis</b>	<b>12</b>
<b>8</b>	<b>Recommendations</b>	<b>13</b>
<b>9</b>	<b>Conclusions</b>	<b>14</b>
<b>Appendices</b>		<b>24</b>
<b>A</b>	<b>Control System Design Criteria: Chief Engineer's Preferences</b>	<b>24</b>
<b>B</b>	<b>Scatter Plots (<code>gplotmatrix</code>) and Heat Map Post NSGA-II Optimisation (Optimisation Result)</b>	<b>25</b>
<b>C</b>	<b>Heat Map Representing Performance Criteria Correlation for Full Factorial Sampling Plan (Knowledge Discovery)</b>	<b>27</b>
<b>D</b>	<b>Scatter Plot and Sampling Post Optimisation (Sustainability Analysis)</b>	<b>28</b>
<b>E</b>	<b>MATLAB Code</b>	<b>30</b>

## List of Figures

1	Full Factorial Sampling . . . . .	5
2	RLH Sampling . . . . .	5
3	Sobol Sampling . . . . .	5
4	Matrix of Scatter Plots ( <code>gplotmatrix</code> ) Representing the Correlation of Performance Criteria Parameters . . . . .	6
5	Parallel Plot of Performance Criteria for Gains from Full Factorial Sampling	7
6	Multi-Objective Optimisation Method Based on the NSGA-II Framework (1)	9
7	Parallel Plot of Performance Criterion Post NSGA-II Optimisation . . . . .	10
8	Sampling Update after NSGA-II Optimisation . . . . .	11
9	Hypervolume Trend Over 175 Iterations . . . . .	11
10	Parallel Plot ( <code>parallelplot</code> ) with Reduced Aggregate Control Effort Energy	12
11	<code>gplotmatrix</code> Scatter Plots Post NSGA-II Optimisation . . . . .	25
12	Heat Map Representing Performance Criteria Correlations Post NSGA-II Optimisation . . . . .	26
13	Heat Map Representing Performance Criteria Correlation for Full Factorial Sampling Plan . . . . .	27
14	Scatter Plots ( <code>gplotmatrix</code> with Reduced Aggregate Control Effort Energy . . . . .	28
15	Sampling Update with Reduced Aggregate Control Effort Energy . . . . .	29

## **List of Tables**

1	Performance Criteria . . . . .	3
2	Control System Design Criteria . . . . .	24

# 1 Multi-objective Optimisation for Engineering Design

Decision Systems for Engineering Design use advanced computational methods to aid decision-making by analyzing data and modeling scenarios (2). These systems simulate, predict, and optimize engineering outcomes, balancing multiple objectives (3). Unlike traditional methods based on heuristics or trial-and-error, decision systems offer a structured, quantifiable, and often automated approach, improving accuracy and expanding the design space (4).

## 1.1 Comparison with Traditional Decision-Making Approaches

Traditional approaches in engineering design typically involve sequential design iterations, expert evaluations, and simplified models that do not always capture the full scope of design complexities (5); they are cost and time ineffective and expert judgement may be misleading at times (6). In contrast, decision systems integrate mathematical models, statistical tools, and algorithm-driven processes to provide a more holistic and detailed analysis, which allows to generate optimal solutions for complex problems (7). This shift allows for a systematic exploration of trade-offs, optimization of multiple objectives simultaneously, and reduction in time and cost by minimizing reliance on physical prototypes. (8) (9)

## 1.2 Applications in Electric Vehicle Design

Multi-objective optimization has become pivotal in the design and development of electric vehicles (EVs) (10), where engineers must often balance performance, cost, sustainability, and safety . Here are five notable applications from the literature:

**Battery Management Systems:** Optimization techniques are used to maximize battery life and performance while minimizing cost and weight (11) (12), crucial for improving the range and efficiency of EVs (13) (14).

**Powertrain Design:** Algorithms help in designing powertrain components (15) that optimize efficiency, torque delivery (16), and energy consumption (17) (18).

**Thermal Management Systems:** Multi-objective approaches are employed to manage the heat generation in batteries (19), electronics (20), and cabin (21), balancing between cooling, performance and energy consumption.

**Charging Infrastructure Deployment:** Decision systems optimize the placement and

type of charging stations (22) to maximize coverage (14) and accessibility while considering installation costs, expected EV adoption rates (23), wind (24), and sustainability (25) (26).

**Material Selection:** Advanced models assist in selecting materials that balance weight, strength, cost, and environmental impact to improve vehicle performance and sustainability (NSGA-II implementation (27)) (28) (29).

### 1.3 Approaches to Multi-Objective Optimisation

The primary methods for solving multi-objective optimization problems in engineering design include the Pareto-front approach (30), weighted sum method (31), and evolutionary algorithms (32). Each of these methods offers distinct mechanisms for handling trade-offs among competing objectives (33) (3):

**Pareto-Front Approach:** This method identifies a set of optimal solutions, known as Pareto optimal solutions, where no objective can be improved without worsening at least one other objective (30). It is particularly useful when the decision-maker wishes to see a range of optimal solutions from which to choose, providing a visual representation of the trade-off landscape (34) (35).

**Weighted Sum Method:** In this approach, different objectives are multiplied by a set of weights that reflect their relative importance, and the products are summed to form a single objective function (31). This method is straightforward and computationally efficient but may not find solutions where trade-offs between objectives are nonlinear or where the Pareto front is non-convex (36) (37).

**Evolutionary Algorithms:** These are stochastic, population-based methods that use mechanisms inspired by biological evolution, such as mutation, crossover, and selection (32) (38). Evolutionary algorithms are highly effective in exploring complex, multi-modal search spaces and can generate a diverse set of solutions across the entire Pareto front (39). However, they may require more computational resources and careful tuning of algorithm parameters. (40) (41)

These methods each offer advantages and limitations, and their selection is guided by the specific application, such as the objectives' nature and the desired solution scope. In engineering design, particularly in the dynamic electric vehicle sector, these approaches enable a more sophisticated, informed, balanced, and efficient decision-making process.

## 2 Problem Formulation

The aim of this project is to optimise the gains of the control system. The multi-objective optimisation is shown in Equation 1

$$Z_n = f(X_i, P_k) \quad (1)$$

where  $Z_n$  is the performance criteria,  $X_i$  the design variables  $K_p$  and  $K_i$ , and  $P_k$  the priority of the goals (where  $k$  ranges from 0 – 3, with  $k = 0$  indicating Low priority,  $k = 1$  indicating Medium priority,  $k = 2$  indication High priority, and  $k = 3$  indicating Hard Constraints).

The Chief Engineer's preferences are given in Table 2 in Appendix A. The performance criteria equations are shown in Table 1

Criterion	Mathematical Expression
Minimize Largest closed-loop pole	$Z_1(X_1, X_2, P_4) < 1$
Maximize Gain Margin	$Z_2(X_1, X_2, P_3) \geq 6 \text{ dB}$
Range Phase Margin	$30^\circ \leq Z_3(X_1, X_2, P_3) \leq 70^\circ$
Minimize Rise Time	$Z_4(X_1, X_2, P_2) \leq 2 \text{ s}$
Minimize Peak Time	$Z_5(X_1, X_2, P_1) \leq 10 \text{ s}$
Minimize Maximum Overshoot	$Z_6(X_1, X_2, P_2) \leq 10\%$
Minimize Maximum Undershoot	$Z_7(X_1, X_2, P_1) \leq 8\%$
Minimize Settling Time	$Z_8(X_1, X_2, P_1) \leq 20 \text{ s}$
Minimize Steady-state Error	$Z_9(X_1, X_2, P_2) \leq 1\%$
Minimize Control Effort	$Z_{10}(X_1, X_2, P_3) \leq 0.67 \text{ MJ}$

Table 1: Performance Criteria

The system must satisfy a hard constraint - largest closed-loop pole magnitude  $|z|$  must be less than 1. The optimization problem can be expressed as:

$$\begin{aligned} \text{minimize} \quad & x \rightarrow z = f(x) \\ \text{subject to} \quad & |z_{max}| < 1, \\ & xK_P, k_I \geq 0 \end{aligned}$$

### 3 Sampling Plan

The space-filling properties of three sampling plans - Full Factorial, Random Latin Hypercube (RLH), and Sobol - were assessed utilising the  $\Phi_q$  metric to evaluate their effectiveness in exploring the control system design space defined by proportional ( $K_P$ ) and integral ( $K_I$ ) gains of a PI controller. The following sampling plans were experimented with in this setup:

**Full Factorial:** A grid-like pattern placing samples at regular intervals across entire range of  $K_P$  and  $K_I$  (42), ensuring complete coverage (computationally intensive) (43). (44)

**Random Latin Hypercube (RLH):** Divides the design space into equally probable segments, providing a good distribution across the range without overlapping samples, advantageous for identifying nonlinear interactions (45) (46) (47). (48)

**Sobol:** Employs a quasi-random sequence that achieves better uniformity than purely random samples, especially beneficial in high-dimensional spaces. (49) (50)

The Euclidean distances (with  $p = 2$  &  $q = 5$ ) between all pairs of samples within the plan is considered. Denoting these unique distances as  $d_\mu$ , and the pairs of samples associated with each distance as  $J_i$ , the  $\Phi_q$  metric for a sampling plan  $X$  is calculated using:

$$\Phi_q(X) = \left( \sum_{j=1}^{\mu} \frac{J_j}{d_j^q} \right)^{1/q}$$

where lower values indicate better space-filling properties. The evaluation produced the following results:

**Full Factorial ( $\Phi_q = 2.6517$ ):** Exhibits a structured distribution with consistent spacing in both dimensions, covering edge cases effectively but might not be as effective for non-linear or irregular objective landscapes. It demonstrates the highest space-filling capability among the sampling strategies, which is essential for a comprehensive exploration of the design space, particularly when optimizing controller gains.

**RLH ( $\Phi_q = 8.0116$ ):** Offers a random yet evenly spread distribution across the variable range, effectively avoiding clustering and provides a good representation of the design space (compared to traditional random sampling).

**Sobol ( $\Phi_q = 3.7972$ ):** Demonstrates a quasi-random spread that is more uniform than simple random sampling, ensuring that all parts of the design space are explored without

clustering, beneficial for ensuring all regions adequately explored (useful in high-dimensional spaces).

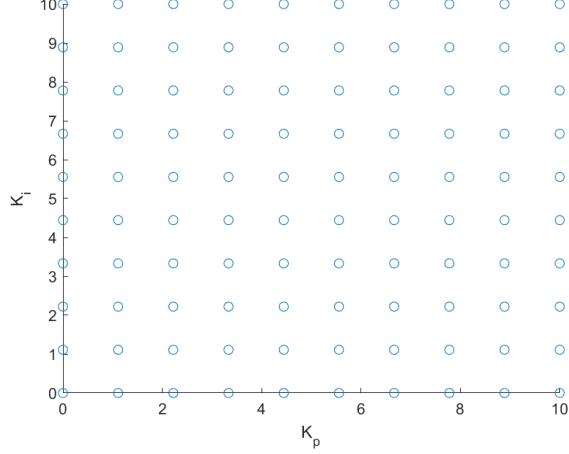


Figure 1: Full Factorial Sampling

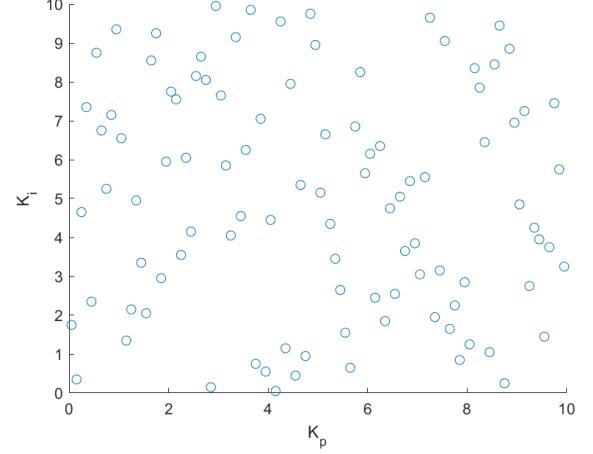


Figure 2: RLH Sampling

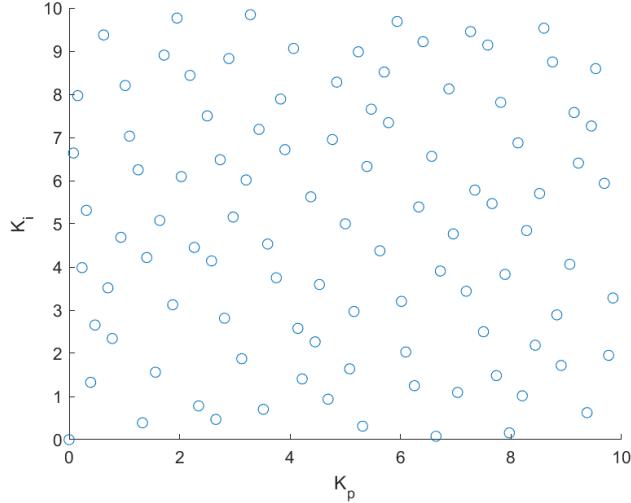


Figure 3: Sobol Sampling

Based on the space-filling properties, the Full Factorial method is recommended for detailed analysis and robust optimization of controller settings. Its ability to explore all combinations of settings ensures that no part of the design space is overlooked, making it ideal for identifying optimal configurations. Although RLH and Sobol are effective, they do not surpass the comprehensive coverage provided by Full Factorial in this context.

## 4 Knowledge Discovery

Figure 4 shows the matrix of scatter plots between all pairs of performance criteria using the  $K_p$  and  $K_i$  values obtained from the Full Factorial sampling plan.

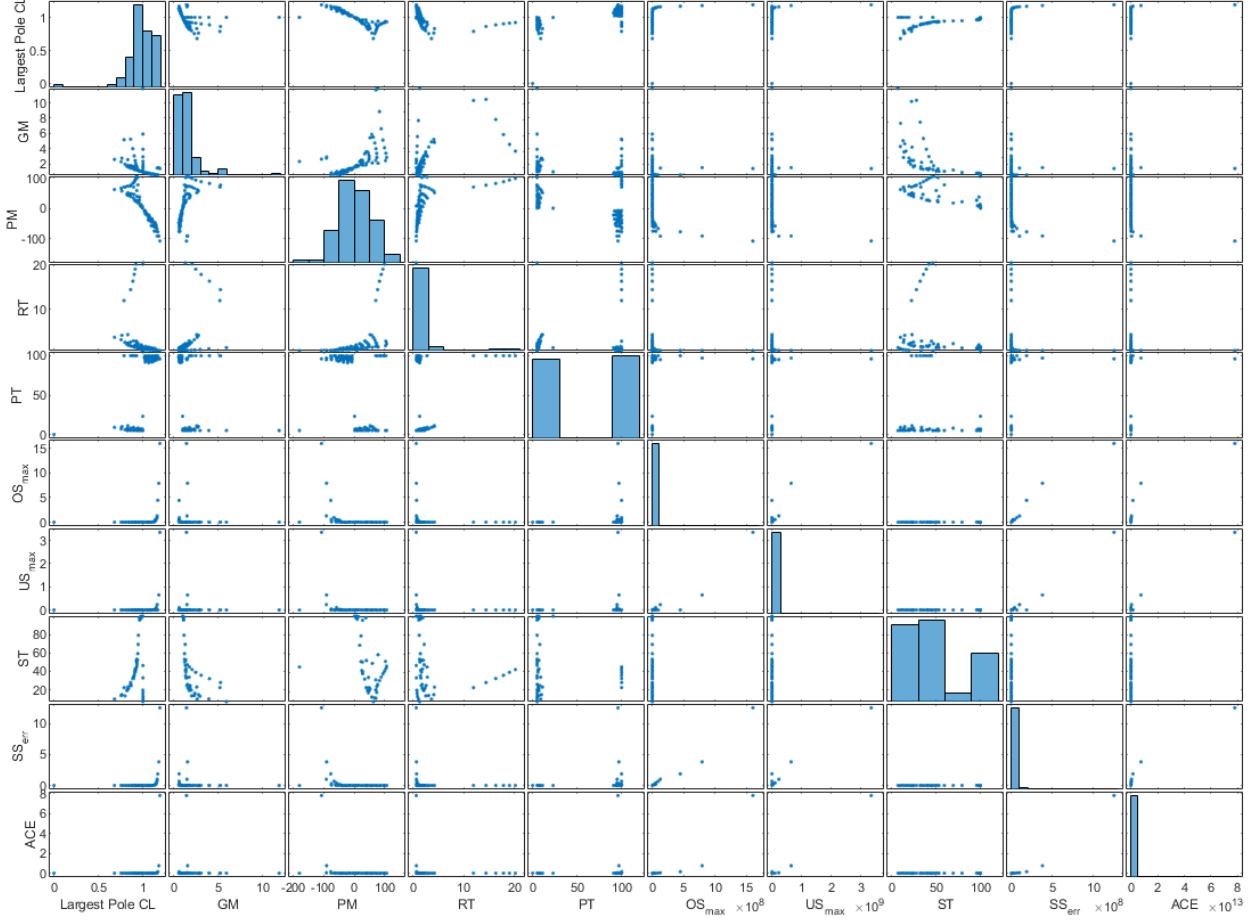


Figure 4: Matrix of Scatter Plots (gplotmatrix) Representing the Correlation of Performance Criteria Parameters

Configurations with lower values of the largest closed-loop pole, indicative of a more stable system, often exhibit higher gain margins and acceptable phase margins. There is a trade-off between rise time and settling time, where faster rise times typically lead to longer settling times due to increased responsiveness causing more oscillations. A pattern emerges where minimizing overshoot often increases undershoot, requiring a delicate balance tailored to the specific needs of the control system.

Figure 5 illustrates the complexity of the system’s behavior using a parallel coordinates plot. Each line represents a set of controller gain settings ( $K_p$  and  $K_i$ ) and their corresponding performance metrics. This visualization underscores how improvements in one area, such as minimizing rise time or overshoot, may necessitate compromises in others, like settling time or steady-state error. Certain clusters within the plot indicate groups of settings that achieve similar performance outcomes, providing a strategic guide for focusing the tuning process on specific regions of the design space.

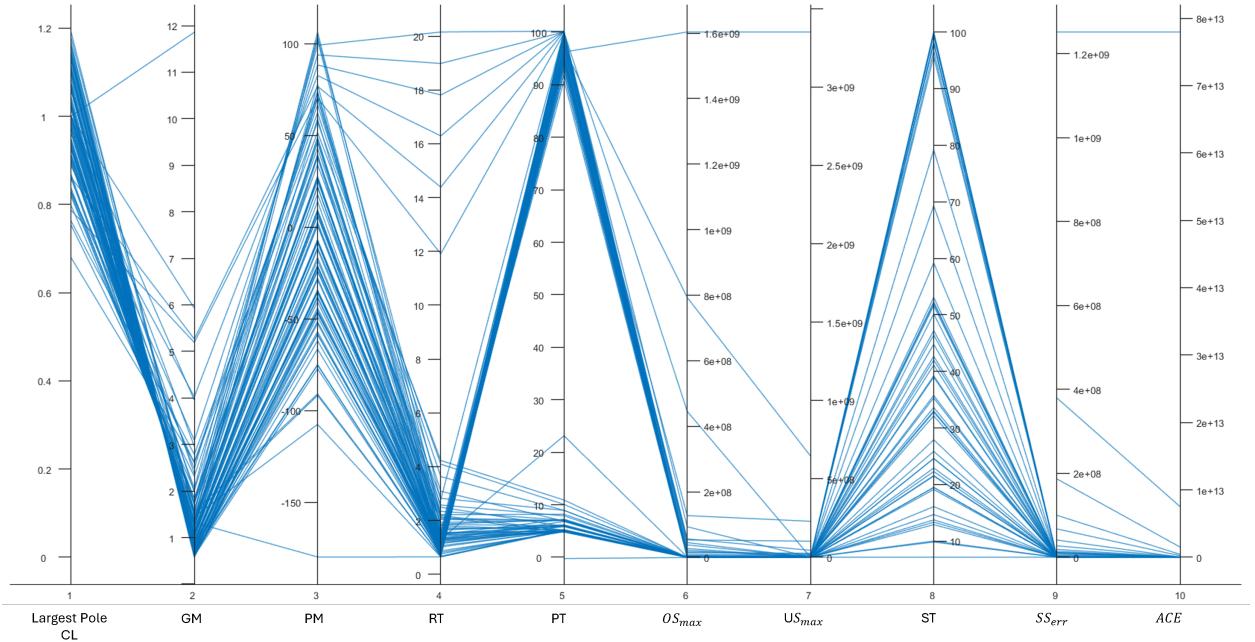


Figure 5: Parallel Plot of Performance Criteria for Gains from Full Factorial Sampling

Certain regions in the design space, particularly those with lower Largest Pole Closed-Loop and moderate Gain Margins, consistently yield stable systems with satisfactory transient and steady-state performance. The data highlights optimal areas where the system effectively balances responsiveness (rise and peak times) with stability (settling time and overshoot/undershoot trade-offs).

The heat map in Figure 13 in Appendix C shows the correlation among the performance criterion in much more detail. For example, the largest closed-loop pole and the settling time possess a strong, positive correlation of 0.85, indicating that both increase or decrease simultaneously. Weaker relationships also exist, such as the correlation between aggregate control effort and the phase margin is 0.02.

## 5 Optimisation Process

The Non Dominated Sorting Genetic Algorithm II (51) was used to solve the multi-objective optimisation of the PI controller gains (52). NSGA-II excels in maintaining a diverse set of solutions along the Pareto front, making it ideal for exploring the trade-offs inherent in optimizing control system parameters (53) (54).

The NSGA-II implementation was as follows (55) (56) (57) (58) (59) (60) (61):

**Initialisation and Pre Processing** The initial population  $P$  was generated using the full factorial sampling plan, representing diverse PI controller settings. The `optimizeControlSystem` function then post-processed and evaluated the designs in the population, adjusting metrics like gain margin from a ratio to decibels and altering the optimization direction where necessary (e.g., from maximizing to minimizing by taking negative values).

**Non-Dominated Sorting and Crowding Distance** Solutions are sorted based on Pareto dominance (`rank_prf`), with less dominated solutions receiving lower ranks. The crowding distance for each solution measures the density of solutions surrounding a particular point in the solution space, helping to maintain diversity by favoring less crowded solutions.

**Binary Tournament Selection:** Binary tournament selection is used, basing choices on fitness (dominance rank and crowding distance) to select solutions for reproduction.

**Crossover, Mutation, Merging and Selecting Next Generation** Crossover (`sbx`) and Polynomial Mutation (`polymut`) are applied to generate new offspring, combining and altering parent characteristics to explore new solution areas. New offspring are combined with the existing population, re-evaluated for fitness, and the best-performing solutions are retained using NSGA-II's elitist strategy.

**Convergence Check: Hypervolume** The hypervolume indicator assesses the convergence and coverage of the Pareto front, indicating algorithm progress or stagnation.

**Iteration and Termination** The optimization loop is run for a predefined number of iterations or until convergence criteria are met. In this case, 125 iterations were specified to

ensure thorough exploration of the solution space.

The detailed NSGA-II algorithm is shown in Figure 6 (1).

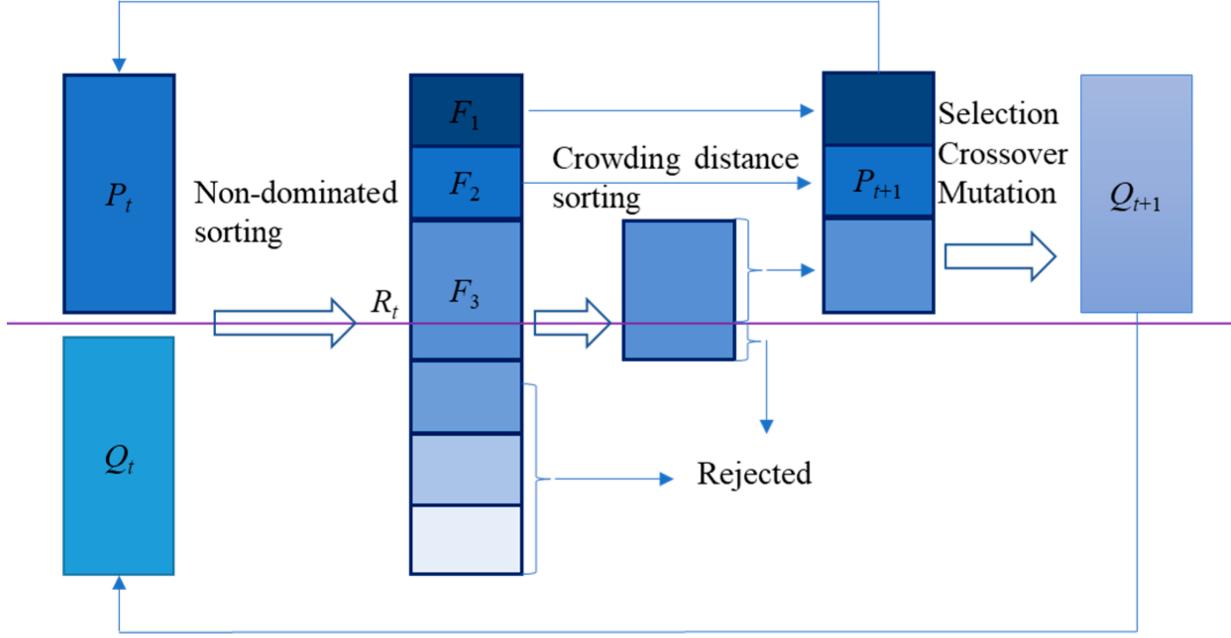


Figure 6: Multi-Objective Optimisation Method Based on the NSGA-II Framework (1)

To efficiently incorporate the preferences of the Chief Engineer, goals were introduced progressively across the optimization iterations -  $\text{Goals} = [0.8 \ 6 \ 20 \ 2 \ 10 \ 10 \ 8 \ 20 \ 1 \ 0.67]$ . This approach enables the algorithm to initially focus on broader targets, then progressively refine the search based on increasingly specific and stringent requirements.

Priority levels were adjusted throughout the optimization process to emphasize different objectives at each stage. The initial 50 iterations concentrated on establishing basic stability with minimal constraints, setting a foundation of feasible solutions. Subsequent 25 iterations introduced and emphasized performance criteria like gain margin and control effort, due to their increased priority. The following 25 iterations incorporated all performance goals, closely aligning with the Chief Engineer's specifications. The final 25 iterations made slight adjustments to balance performance criteria with control effort, enhancing practical applicability.

NSGA-II is used for many applications, including electric vehicle energy management (62) (63) and charging site selection (64).

## 6 Optimisation Results

Figure 11 in Appendix B shows insights into the correlations and distributions of the performance criteria post NSGA-II optimisation.

Figure 7 shows the parallel plot of the optimised samples. Note that the negative of the gain margin is represented on the parallel plot due to its negative inversion as part of the `optimizeControlSystem` function. The phase margin is normalized by subtracting 50 and then taking the absolute value.

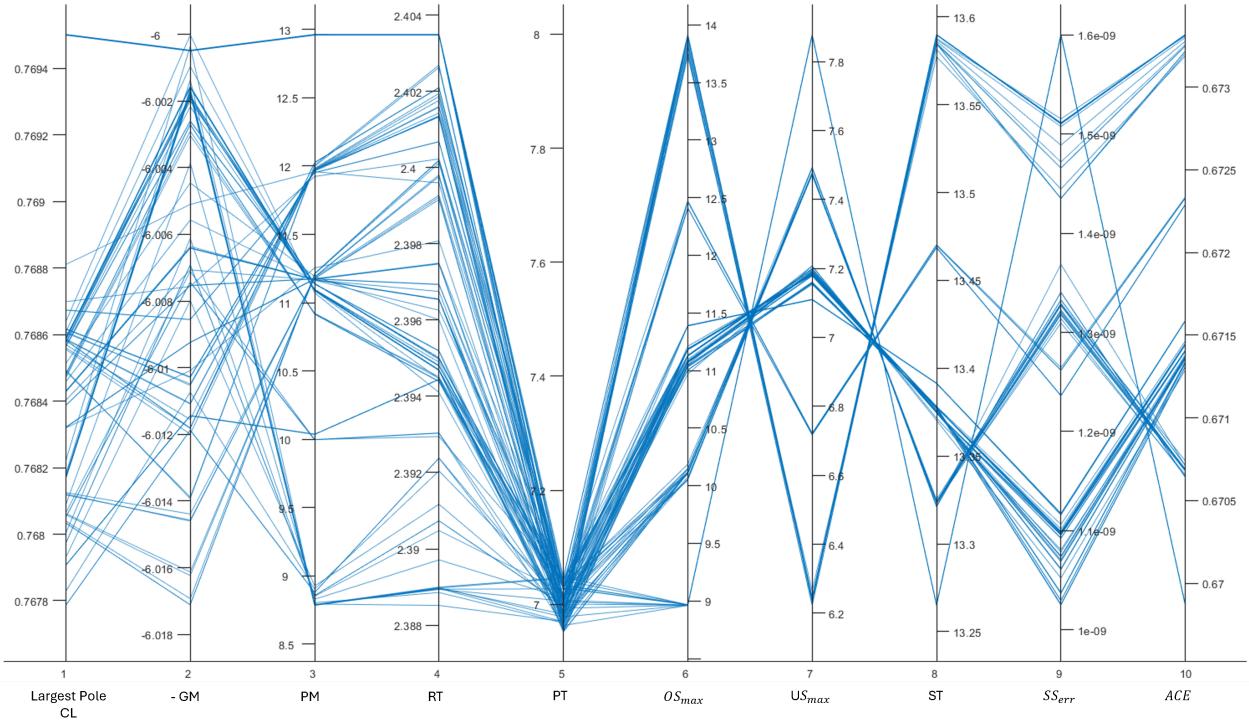


Figure 7: Parallel Plot of Performance Criterion Post NSGA-II Optimisation

Figure 8 shows the optimised  $K_P$  and  $K_I$  values obtained post NSGA-II application.

Most criteria are met except for rise time, which could be improved but would compromise control effort. The optimal performance evaluation for  $Z$  and corresponding gains are:

$$Z_{Optimal} = [0.77, -6.00, 12.96, 2.40, 7.96, 7.88, 13.27, 1.60 \times 10^{-9}, 0.67]$$

$$[K_P, K_I] = [0.397, 0.253]$$

The solution achieved optimal fit (100% satisfaction) for most of the chief engineer's preferences, except for the rise time, which **exceeds target by 0.4 seconds**. As the optimization process provided Pareto-optimal solution, satisfying the rise time criteria would compromise another criterion, specifically the aggregate control effort (ACE).

Figure 9 illustrates the hypervolume's progression over 125 iterations. The reference point, set at the maximum of  $Z_{eval}$ , represents the worst-case scenario for each objective, defining the scope of the hypervolume calculation.

Early in the optimization, high initial hypervolume values indicate extensive coverage of the objective space, approaching the worst-case reference point.

A noticeable decrease in hypervolume after around iteration 80 indicates a significant refinement in the Pareto front, reflecting convergence towards a more optimal set of solution. Post-drop, the hypervolume stabilizes, indicating that further iterations do not significantly enhance the coverage of the Pareto front. Incorporating mechanisms such as niching [REFERENCE] may prevent convergence on local optima.

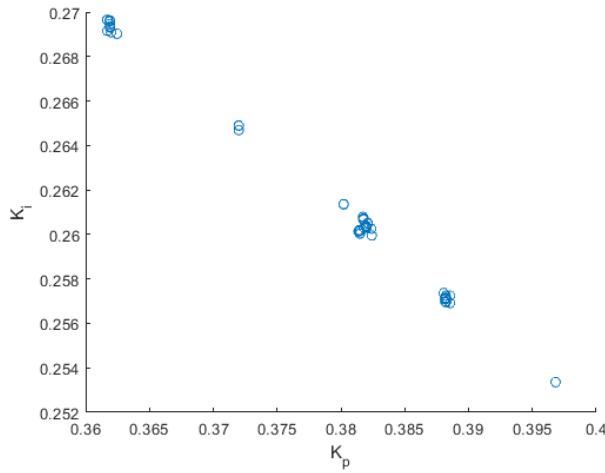


Figure 8: Sampling Update after NSGA-II Optimisation

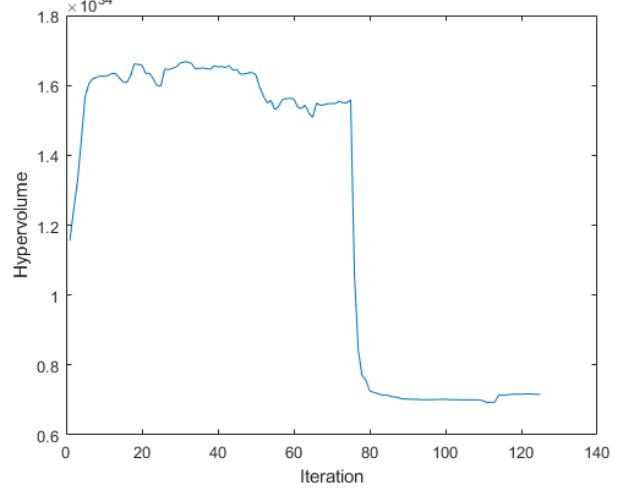


Figure 9: Hypervolume Trend Over 175 Iterations

The heatmap in Figure 12 in Appendix B shows the performance criteria correlations post optimisation, showing the highly inversely proportional relationship ( $-0.94$ ) between the rise time and ACE, which prevents complete satisfaction of the preferences.

## 7 Sustainability Analysis

Figure 10 shows the parallel plot with the control effort energy goal reduced to  $0.63MJ$ .

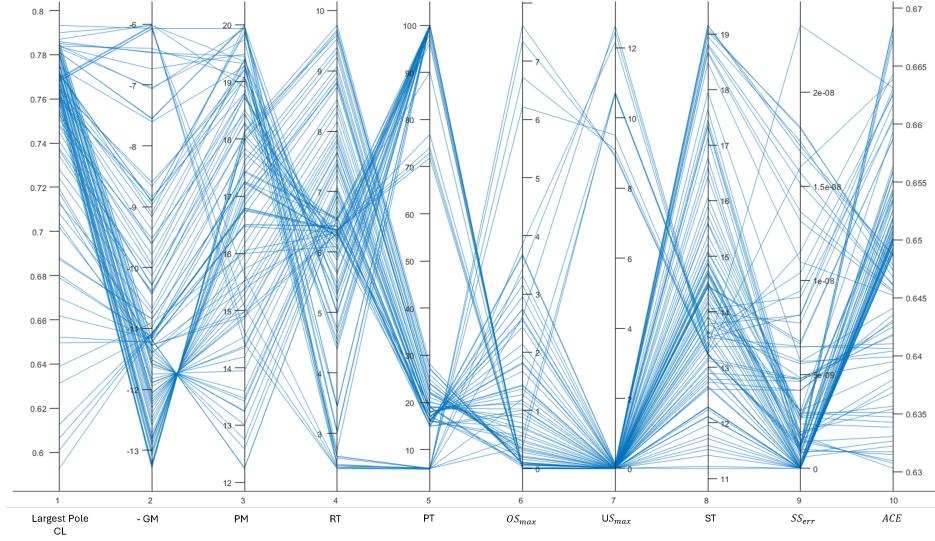


Figure 10: Parallel Plot (parallelplot) with Reduced Aggregate Control Effort Energy

The targeted reduction in control effort represents a high sustainability goal aimed at minimizing energy usage. The optimal performance evaluation for  $Z$  and corresponding gains when the control efforts goal is reduced to  $0.63MJ$  are:

$$Z_{Optimal} = [0.77, -12.25, 15.39, 7.94, 19, 1.11, 0.012, 13.77, 5.68 \times 10^{-9}, 0.63]$$

$$[K_P, K_I] = [2.05 \times 10^{-4}, 0.12]$$

As anticipated by the trends shown in the heat maps in Appendix B and C, tightening the control effort resulted in increased rise time and peak time (beyond the preference), impacting the system's responsiveness and operational agility, increasing inertia and delaying responses, due to longer rise times and reduced natural frequency from peak time errors.

Reducing control effort often results in longer rise and peak times; a trade-off where lower energy use impacts responsiveness (65). High-efficiency motors and regenerative systems can mitigate this by enhancing performance at lower energy levels and recovering energy during operation. Incorporating energy-aware control algorithms that balance system dynamics with energy efficiency optimizes performance without compromising effectiveness.

## 8 Recommendations

Based on an extensive analysis of design variables and performance criteria, the recommended PI controller settings for the control system for the chief engineer's preferences are:

$$[K_P, K_I]_{\text{Base Criteria}} = [0.397, 0.253]$$

The optimisation results satisfy most of the desired chief engineer's criteria, except for the rise time, which is violated by 0.4 seconds (20% higher than target).

The recommended PI controller settings for the sustainable preference criteria (reduced control effort goal of  $0.63MJ$ ) are:

$$[K_P, K_I]_{\text{Sustainable Criteria}} = [2.05 \times 10^{-4}, 0.12]$$

The optimisation results for the sustainable preference criteria satisfy the reduced control effort goal at the expense of significantly deviating from the rise and peak time targets (by approximately 400% and 190%).

Since the results are Pareto-optimal, satisfying the rise and/or peak time criteria would compromise other criterion, specifically the aggregate control effort (ACE). Possible methods of system enhancements to reduce rise and peak times include the use of faster, more efficient (high-performance) actuators (66) to improve response times without increasing energy consumption (67). Use of advanced controller architecture such as implementing sophisticated control strategies like Model Predictive Control (68) (69) or adaptive control (70) can enhance system performance. Using feedforward control in conjunction with the PI controller (71) (72) can enhance the system response as integrating mechanisms that predict changes based on reference inputs can lessen the feedback control's lead, thus improving rise time. Using lighter materials, such as carbon fiber in electric vehicles (73), could decrease the energy needed for movements, enhancing overall efficiency (74). Compromising on the rise or peak time could allow to achieve the chief engineer's preferences (in Appendix A).

The PI controller settings should be tuned based on the application or instance. For instance, if the application requires faster response times, the  $[K_P, K_I]_{\text{Base Criteria}}$  settings may be used at the expense of higher energy consumption. Optimal values should be determined through additional real-time simulations and field tests that take into account the various operational conditions the vehicle may face.

## 9 Conclusions

The report explores the optimisation of PI controller gains for an electric vehicle's feedback system, aiming to meet the chief engineer's performance criteria. The methodologies and tools used in this analysis are directly applicable to the broader vehicle propulsion challenges faced by your company.

The multi-objective optimization approach, specifically using the NSGA-II algorithm, has shown how varying PI controller settings can balance multiple performance targets, including energy efficiency and transient response characteristics. This methodology is particularly beneficial for electric vehicles where the interplay between energy management, performance, and sustainability is crucial.

For your vehicle propulsion problem, the same optimization techniques can help manage trade-offs between propulsion power, energy consumption, and overall vehicle dynamics. By extending these techniques, we can ensure that the propulsion system not only meets performance goals but also adheres to sustainability standards.

Additionally, two other decision systems tools that would be beneficial in complementing this optimization process include (would be incorporated into the **Goals** array as shown in Appendix E):

- **Agent-Based Modeling (ABM):** This tool can simulate the interactions of autonomous agents (components of the propulsion system (75)) to predict complex behaviors and outcomes in different scenarios. ABM can provide insights into how changes in one part of the system might impact overall vehicle performance (76). (77) (78) (79)
- **System Dynamics Modeling:** This involves constructing and analyzing simulation models to understand the behavior of complex systems over time (80). In the context of electric vehicles (81), this can help in forecasting how changes in design or operation affect vehicle reliability (82), sustainability (83), and compliance with regulations over its life cycle. (84) (85)

Incorporating these tools alongside the optimization methodologies demonstrated in this study will provide a robust framework for decision-making that enhances design quality, reduces time to market, and ensures that the vehicle meets or exceeds all performance and sustainability criteria.

## References

- [1] R. Jiang, S. Ci, D. Liu, X. Cheng, and Z. Pan, “A hybrid multi-objective optimization method based on nsga-ii algorithm and entropy weighted topsis for lightweight design of dump truck carriage,” *Machines*, vol. 9, no. 8, 2021.
- [2] G. A. Hazelrigg, “A framework for decision-based engineering design,” 1998.
- [3] N. Gunantara, “A review of multi-objective optimization: Methods and its applications,” *Cogent Engineering*, vol. 5, no. 1, p. 1502242, 2018.
- [4] D. M. Buede and W. D. Miller, *The engineering design of systems: models and methods*. John Wiley & Sons, 2024.
- [5] J. G. Johnson and J. R. Busemeyer, *Computational Models of Decision Making*, p. 499–526. Cambridge Handbooks in Psychology, Cambridge University Press, 2023.
- [6] W. Chen, C. Hoyle, and H. J. Wassenaar, *Decision Theory in Engineering Design*, pp. 13–34. London: Springer London, 2013.
- [7] Read "Theoretical Foundations for Decision Making in Engineering Design" at [NAP.edu](http://NAP.edu).
- [8] A. Marketing, “Exploring the advantages between traditional and bim design,” 01 2024.
- [9] “Dynamic decision making in engineering system design: A deep q-learning approach.”
- [10] A. Mansouri, N. Smairi, and H. Trabelsi, “Multi-objective optimization of an in-wheel electric vehicle motor,” *International Journal of Applied Electromagnetics and Mechanics*, vol. 50, no. 3, pp. 449–465, 2016.
- [11] R. Das, Y. Wang, K. Busawon, G. Putrus, and M. Neaimeh, “Real-time multi-objective optimisation for electric vehicle charging management,” *Journal of Cleaner Production*, vol. 292, p. 126066, 2021.
- [12] Z. Song, J. Li, X. Han, L. Xu, L. Lu, M. Ouyang, and H. Hofmann, “Multi-objective optimization of a semi-active battery/supercapacitor energy storage system for electric vehicles,” *Applied Energy*, vol. 135, pp. 212–224, 2014.

- [13] A. Panday and H. O. Bansal, “Multi-objective optimization in battery selection for hybrid electric vehicle applications.,” *Journal of Electrical Systems*, vol. 12, no. 2, 2016.
- [14] W. Vaz, A. K. Nandi, R. G. Landers, and U. O. Koylu, “Electric vehicle range prediction for constant speed trip using multi-objective optimization,” *Journal of Power Sources*, vol. 275, pp. 435–446, 2015.
- [15] K. Kwon, M. Seo, and S. Min, “Multi-objective optimization of powertrain components for electric vehicles using a two-stage analysis model,” *International Journal of Automotive Technology*, vol. 21, pp. 1495–1505, 2020.
- [16] K. Kwon, M. Seo, and S. Min, “Efficient multi-objective optimization of gear ratios and motor torque distribution for electric vehicles with two-motor and two-speed powertrain system,” *Applied Energy*, vol. 259, p. 114190, 2020.
- [17] S. F. da Silva, J. J. Eckert, F. L. Silva, L. C. Silva, and F. G. Dedini, “Multi-objective optimization design and control of plug-in hybrid electric vehicle powertrain for minimization of energy consumption, exhaust emissions and battery degradation,” *Energy Conversion and Management*, vol. 234, p. 113909, 2021.
- [18] X. Zhou, D. Qin, and J. Hu, “Multi-objective optimization design and performance evaluation for plug-in hybrid electric vehicle powertrains,” *Applied Energy*, vol. 208, pp. 1608–1625, 2017.
- [19] W. Li, X. Peng, M. Xiao, A. Garg, and L. Gao, “Multi-objective design optimization for mini-channel cooling battery thermal management system in an electric vehicle,” *International Journal of Energy Research*, vol. 43, no. 8, pp. 3668–3680, 2019.
- [20] S. Su, W. Li, Y. Li, A. Garg, L. Gao, and Q. Zhou, “Multi-objective design optimization of battery thermal management system for electric vehicles,” *Applied Thermal Engineering*, vol. 196, p. 117235, 2021.
- [21] I. Cvok, I. Ratković, and J. Deur, “Multi-objective optimisation-based design of an electric vehicle cabin heating control system for improved thermal comfort and driving range,” *Energies*, vol. 14, no. 4, p. 1203, 2021.

- [22] K. Liu, K. Li, H. Ma, J. Zhang, and Q. Peng, “Multi-objective optimization of charging patterns for lithium-ion battery management,” *Energy Conversion and Management*, vol. 159, pp. 151–162, 2018.
- [23] R. Wang, J. Mu, Z. Sun, J. Wang, and A. Hu, “Nsga-ii multi-objective optimization regional electricity price model for electric vehicle charging based on travel law,” *Energy Reports*, vol. 7, pp. 1495–1503, 2021.
- [24] B. Sun, “A multi-objective optimization model for fast electric vehicle charging stations with wind, pv power and energy storage,” *Journal of Cleaner Production*, vol. 288, p. 125564, 2021.
- [25] S. Barakat, A. I. Osman, E. Tag-Eldin, A. A. Telba, H. M. A. Mageed, and M. Samy, “Achieving green mobility: Multi-objective optimization for sustainable electric vehicle charging,” *Energy Strategy Reviews*, vol. 53, p. 101351, 2024.
- [26] R. Das, Y. Wang, G. Putrus, R. Kotter, M. Marzband, B. Herteleer, and J. Warmerdam, “Multi-objective techno-economic-environmental optimisation of electric vehicle for energy services,” *Applied Energy*, vol. 257, p. 113965, 2020.
- [27] P. Zhang, Y. Qian, and Q. Qian, “Multi-objective optimization for materials design with improved nsga-ii,” *Materials Today Communications*, vol. 28, p. 102709, 2021.
- [28] M. Ashby, “Multi-objective optimization in material design and selection,” *Acta materialia*, vol. 48, no. 1, pp. 359–369, 2000.
- [29] A. J. Goupee and S. S. Vel, “Multi-objective optimization of functionally graded materials with temperature-dependent material properties,” *Materials & design*, vol. 28, no. 6, pp. 1861–1879, 2007.
- [30] P. Ngatchou, A. Zarei, and A. El-Sharkawi, “Pareto multi objective optimization,” in *Proceedings of the 13th international conference on, intelligent systems application to power systems*, pp. 84–91, IEEE, 2005.
- [31] R. T. Marler and J. S. Arora, “The weighted sum method for multi-objective optimization: new insights,” *Structural and multidisciplinary optimization*, vol. 41, pp. 853–862, 2010.

- [32] K. Deb, “Multi-objective optimisation using evolutionary algorithms: an introduction,” in *Multi-objective evolutionary optimisation for product design and manufacturing*, pp. 3–34, Springer, 2011.
- [33] I. Giagkiozis and P. J. Fleming, “Methods for multi-objective optimization: An analysis,” *Information Sciences*, vol. 293, pp. 338–350, 2015.
- [34] Ö. Ciftcioglu and M. S. Bittermann, “Adaptive formation of pareto front in evolutionary multi-objective optimization,” *Evolutionary computation*, pp. 417–444, 2009.
- [35] “Pareto front,” 11 2023.
- [36] L. Ng, N. Chemmangattuvalappil, V. Dev, and M. Eden, “Chapter 1 - mathematical principles of chemical product design and strategies,” in *Tools For Chemical Product Design* (M. Martín, M. R. Eden, and N. G. Chemmangattuvalappil, eds.), vol. 39 of *Computer Aided Chemical Engineering*, pp. 3–43, Elsevier, 2016.
- [37] I. Kim and O. de Weck, “Adaptive weighted sum method for multiobjective optimization,” *10th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, 08 2004.
- [38] A. Konak, D. W. Coit, and A. E. Smith, “Multi-objective optimization using genetic algorithms: A tutorial,” *Reliability engineering & system safety*, vol. 91, no. 9, pp. 992–1007, 2006.
- [39] A. Zhou, B.-Y. Qu, H. Li, S.-Z. Zhao, P. N. Suganthan, and Q. Zhang, “Multiobjective evolutionary algorithms: A survey of the state of the art,” *Swarm and Evolutionary Computation*, vol. 1, no. 1, pp. 32–49, 2011.
- [40] W. Contributors, “Multi-objective optimization,” 07 2019.
- [41] K. Deb, *Multi-objective optimization using evolutionary algorithms*, vol. 16. John Wiley & Sons, 2001.
- [42] H. M. Hasanien and S. Muyeen, “A taguchi approach for optimum design of proportional-integral controllers in cascaded control scheme,” *IEEE Transactions on Power Systems*, vol. 28, no. 2, pp. 1636–1644, 2012.

- [43] P. Sedgwick, “Randomised controlled trials with full factorial designs,” *BMJ*, vol. 345, pp. e5114–e5114, 08 2012.
- [44] N. Yamakawa, “Random combined fractional factorial designs 1 ; theory and application of the random combined fractional factorial designs (racoffd),” *Bulletin of Mathematical Statistics*, vol. 11, pp. 1–38, 03 1965.
- [45] W. Contributors, “Latin hypercube sampling,” 10 2019.
- [46] Stephanie, “Latin hypercube sampling: Simple definition,” 01 2018.
- [47] “Latin hypercube sample - matlab lhsdesign - mathworks united kingdom.”
- [48] A. Parnianifard, A. Azfanizam, M. Ariffin, M. Ismail, M. R. Maghami, and C. Gomes, “Kriging and latin hypercube sampling assisted simulation optimization in optimal design of pid controller for speed control of dc motor,” *Journal of Computational and Theoretical Nanoscience*, vol. 15, no. 5, pp. 1471–1479, 2018.
- [49] “Sobol sequence,” 05 2023.
- [50] “Understanding sobol sequences.”
- [51] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, “A fast and elitist multiobjective genetic algorithm: Nsga-ii,” *IEEE transactions on evolutionary computation*, vol. 6, no. 2, pp. 182–197, 2002.
- [52] K. Deb, S. Agrawal, A. Pratap, and T. Meyarivan, “A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: Nsga-ii,” in *Parallel Problem Solving from Nature PPSN VI: 6th International Conference Paris, France, September 18–20, 2000 Proceedings* 6, pp. 849–858, Springer, 2000.
- [53] K. Deb, A. Pratap, S. Agarwal, T. Meyarivan, and A. Fast, “Nsga-ii,” *IEEE transactions on evolutionary computation*, vol. 6, no. 2, pp. 182–197, 2002.
- [54] S. Verma, M. Pant, and V. Snasel, “A comprehensive review on nsga-ii for multi-objective combinatorial optimization problems,” *IEEE access*, vol. 9, pp. 57757–57791, 2021.

- [55] V. De Buck, C. A. M. López, P. Nimmemeers, I. Hashem, and J. Van Impe, “Multi-objective optimisation of chemical processes via improved genetic algorithms: A novel trade-off and termination criterion,” in *29th European Symposium on Computer Aided Process Engineering* (A. A. Kiss, E. Zondervan, R. Lakerveld, and L. Özkan, eds.), vol. 46 of *Computer Aided Chemical Engineering*, pp. 613–618, Elsevier, 2019.
- [56] M. W. Mkaouer and M. Kessentini, “Chapter four - model transformation using multi-objective optimization,” vol. 92 of *Advances in Computers*, pp. 161–202, Elsevier, 2014.
- [57] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, “A fast and elitist multiobjective genetic algorithm: Nsga-ii,” *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, 2002.
- [58] “pymoo - nsga-ii: Non-dominated sorting genetic algorithm.”
- [59] “Nsga-2 multi-objective genetic algorithm. anyone could give me a ”simple explanation”?.”
- [60] “Non dominated sorting genetic algorithm (nsga-ii) — pagmo 2.19.0 documentation.”
- [61] P. Calle, “Nsga-ii explained!,” 10 2017.
- [62] Y. Li, S. Wang, X. Duan, S. Liu, J. Liu, and S. Hu, “Multi-objective energy management for atkinson cycle engine and series hybrid electric vehicle based on evolutionary nsga-ii algorithm using digital twins,” *Energy Conversion and Management*, vol. 230, p. 113788, 2021.
- [63] T. Deng, C. Lin, J. Luo, and B. Chen, “Nsga-ii multi-objectives optimization algorithm for energy management control of hybrid electric vehicle,” *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, vol. 233, no. 4, pp. 1023–1034, 2019.
- [64] H. Zhang and F. Shi, “A multi-objective site selection of electric vehicle charging station based on nsga-ii,” *International Journal of Industrial Engineering Computations*, vol. 15, no. 1, pp. 293–306, 2024.

- [65] M. Sheng, Y. Li, X. Wang, J. Li, and Y. Shi, “Energy efficiency and delay tradeoff in device-to-device communications underlaying cellular networks,” *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 1, pp. 92–106, 2015.
- [66] H. Jin, X. Gao, K. Ren, J. Liu, L. Qiao, M. Liu, W. Chen, Y. He, S. Dong, Z. Xu, *et al.*, “Review on piezoelectric actuators based on high-performance piezoelectric materials,” *IEEE Transactions on ultrasonics, ferroelectrics, and frequency control*, vol. 69, no. 11, pp. 3057–3069, 2022.
- [67] H. Tang, Y. Chen, and A. Zhou, “Actuator fault-tolerant control for four-wheel-drive-by-wire electric vehicle,” *IEEE transactions on transportation electrification*, vol. 8, no. 2, pp. 2361–2373, 2021.
- [68] D. Tavernini, M. Metzler, P. Gruber, and A. Sorniotti, “Explicit nonlinear model predictive control for electric vehicle traction control,” *IEEE Transactions on Control Systems Technology*, vol. 27, no. 4, pp. 1438–1451, 2018.
- [69] R. Halvgaard, N. K. Poulsen, H. Madsen, J. B. Jørgensen, F. Marra, and D. E. M. Bondy, “Electric vehicle charge planning using economic model predictive control,” in *2012 IEEE International Electric Vehicle Conference*, pp. 1–6, IEEE, 2012.
- [70] R. Sacks and C. Cox, “Design of an adaptive control system for a hybrid electric vehicle,” in *IEEE SMC’99 Conference Proceedings. 1999 IEEE International Conference on Systems, Man, and Cybernetics (Cat. No. 99CH37028)*, vol. 6, pp. 1000–1005, IEEE, 1999.
- [71] E. Zerdali and R. Demir, “Speed-sensorless predictive torque controlled induction motor drive with feed-forward control of load torque for electric vehicle applications,” *Turkish Journal of Electrical Engineering and Computer Sciences*, vol. 29, no. 1, pp. 223–240, 2021.
- [72] G. Kaiser, F. Holzmann, B. Chretien, M. Korte, and H. Werner, “Torque vectoring with a feedback and feed forward controller-applied to a through the road hybrid electric vehicle,” in *2011 IEEE intelligent vehicles symposium (IV)*, pp. 448–453, IEEE, 2011.

- [73] M. Anwar, S. I. Cahyono, W. R. Wijang, and K. Diharjo, “Application of carbon fiber-based composite for electric vehicle,” *Advanced Materials Research*, vol. 896, pp. 574–577, 2014.
- [74] D. Pandey, K. Sambath Kumar, L. N. Henderson, G. Suarez, P. Vega, H. R. Salvador, L. Roberson, and J. Thomas, “Energized composites for electric vehicles: A dual function energy-storing supercapacitor-based carbon fiber composite for the body panels,” *Small*, vol. 18, no. 9, p. 2107053, 2022.
- [75] “Optimising electric vehicle propulsion system efficiency and sustainability,” 05 2023.
- [76] C. M. Macal and M. J. North, “Tutorial on agent-based modelling and simulation,” *Journal of Simulation*, vol. 4, no. 3, pp. 151–162, 2010.
- [77] A. Kangur, W. Jager, R. Verbrugge, and M. Bockarjova, “An agent-based model for diffusion of electric vehicles,” *Journal of Environmental Psychology*, vol. 52, pp. 166–182, 2017.
- [78] M. Pagani, W. Korosec, N. Chokani, and R. S. Abhari, “User behaviour and electric vehicle charging infrastructure: An agent-based model assessment,” *Applied Energy*, vol. 254, p. 113680, 2019.
- [79] C. Delcea and N. Chirita, “Exploring the applications of agent-based modeling in transportation,” *Applied Sciences*, vol. 13, no. 17, 2023.
- [80] R. G. Coyle, “System dynamics modelling: a practical approach,” *Journal of the Operational Research Society*, vol. 48, no. 5, pp. 544–544, 1997.
- [81] D. Khan, K. Mandal, A. K. Ray, and A. El Aroudi, “A unified switched nonlinear dynamic model of an electric vehicle for performance evaluation,” *Electronics*, vol. 12, no. 14, 2023.
- [82] P. Basirat, H. Fazlollahtabar, and I. Mahdavi, “System dynamics meta-modelling for reliability considerations in maintenance,” *International Journal of Process Management and Benchmarking*, vol. 3, no. 2, pp. 136–153, 2013.

- [83] N. C. Onat, M. Kucukvar, O. Tatari, and G. Egilmez, “Integration of system dynamics approach toward deepening and broadening the life cycle sustainability assessment framework: a case for electric vehicles,” *The International Journal of Life Cycle Assessment*, vol. 21, pp. 1009–1034, 2016.
- [84] Y. Xiang, H. Zhou, W. Yang, J. Liu, Y. Niu, and J. Guo, “Scale evolution of electric vehicles: A system dynamics approach,” *Ieee Access*, vol. 5, pp. 8859–8868, 2017.
- [85] E. S. I. Group, “Optimizing electric vehicle performance and defining its character,” 04 2021.

# Appendices

## A Control System Design Criteria: Chief Engineer's Preferences

Criterion	Direction	Goal	Priority
Largest closed-loop pole	Minimise	< 1	Hard constraint
Gain margin	Maximise	6 dB	High
Phase margin	Range	$\geq 30^\circ$ and $\leq 70^\circ$	High
Rise time	Minimise	2 s	Moderate
Peak time	Minimise	10 s	Low
Maximum overshoot	Minimise	10%	Moderate
Maximum undershoot	Minimise	8%	Low
Settling time	Minimise	20 s	Low
Steady-state error	Minimise	1%	Moderate
Control effort	Minimise	0.67 MJ	High

Table 2: Control System Design Criteria

## B Scatter Plots (gplotmatrix) and Heat Map Post NSGA-II Optimisation (Optimisation Result)

Figure 11 shows insights into the correlations and distributions of the performance criteria post NSGA-II optimisation.

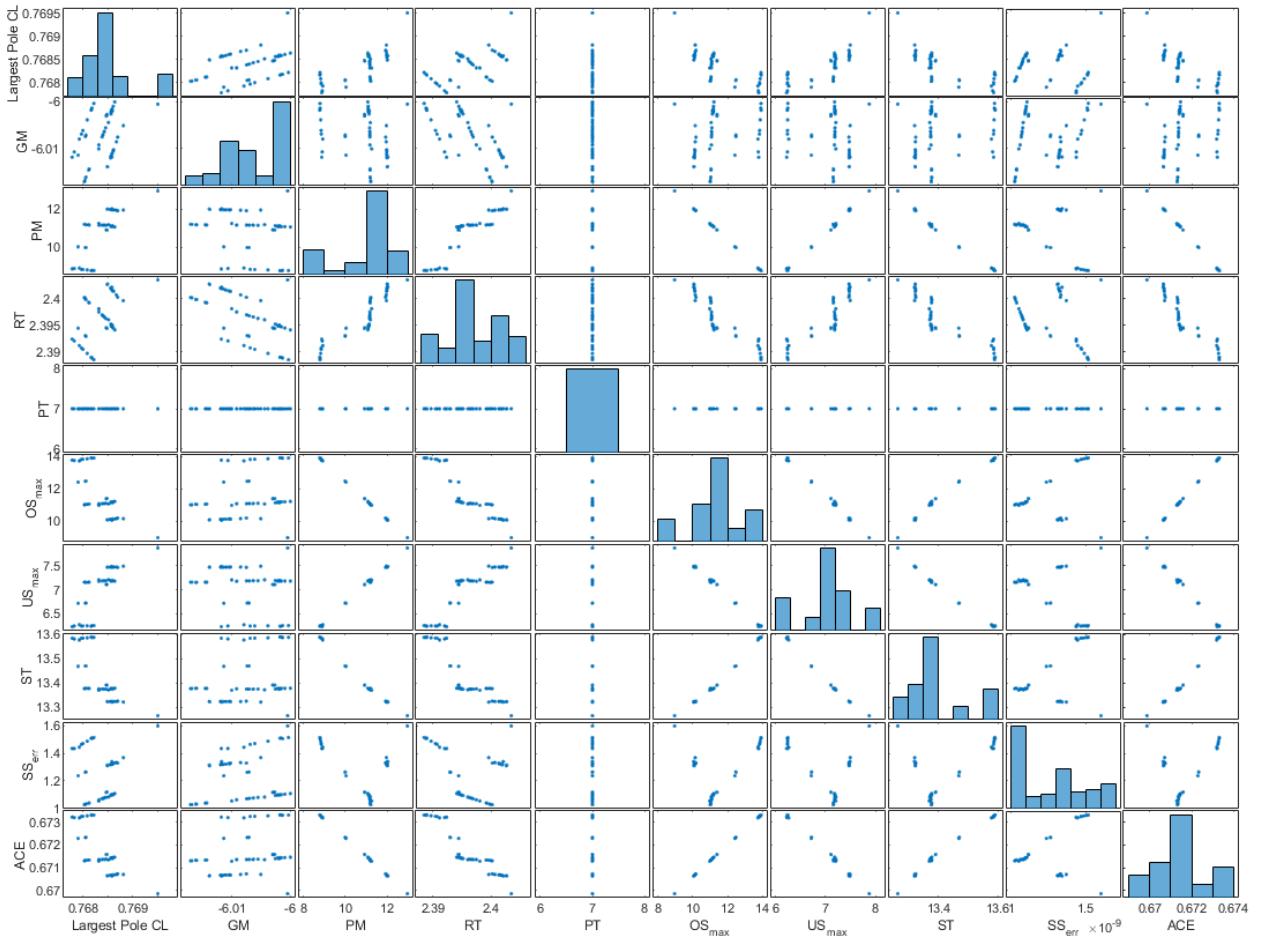


Figure 11: gplotmatrix Scatter Plots Post NSGA-II Optimisation

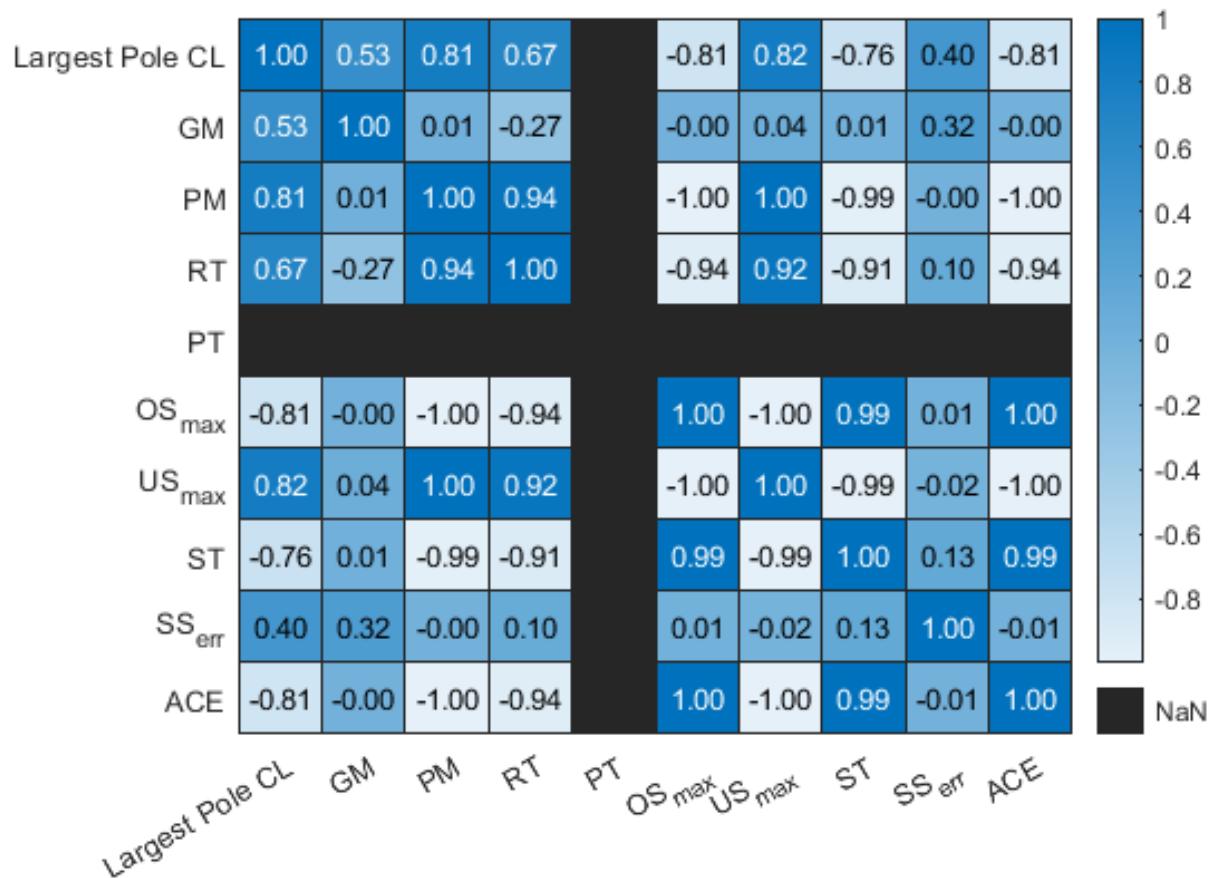


Figure 12: Heat Map Representing Performance Criteria Correlations Post NSGA-II Optimisation

## C Heat Map Representing Performance Criteria Correlation for Full Factorial Sampling Plan (Knowledge Discovery)

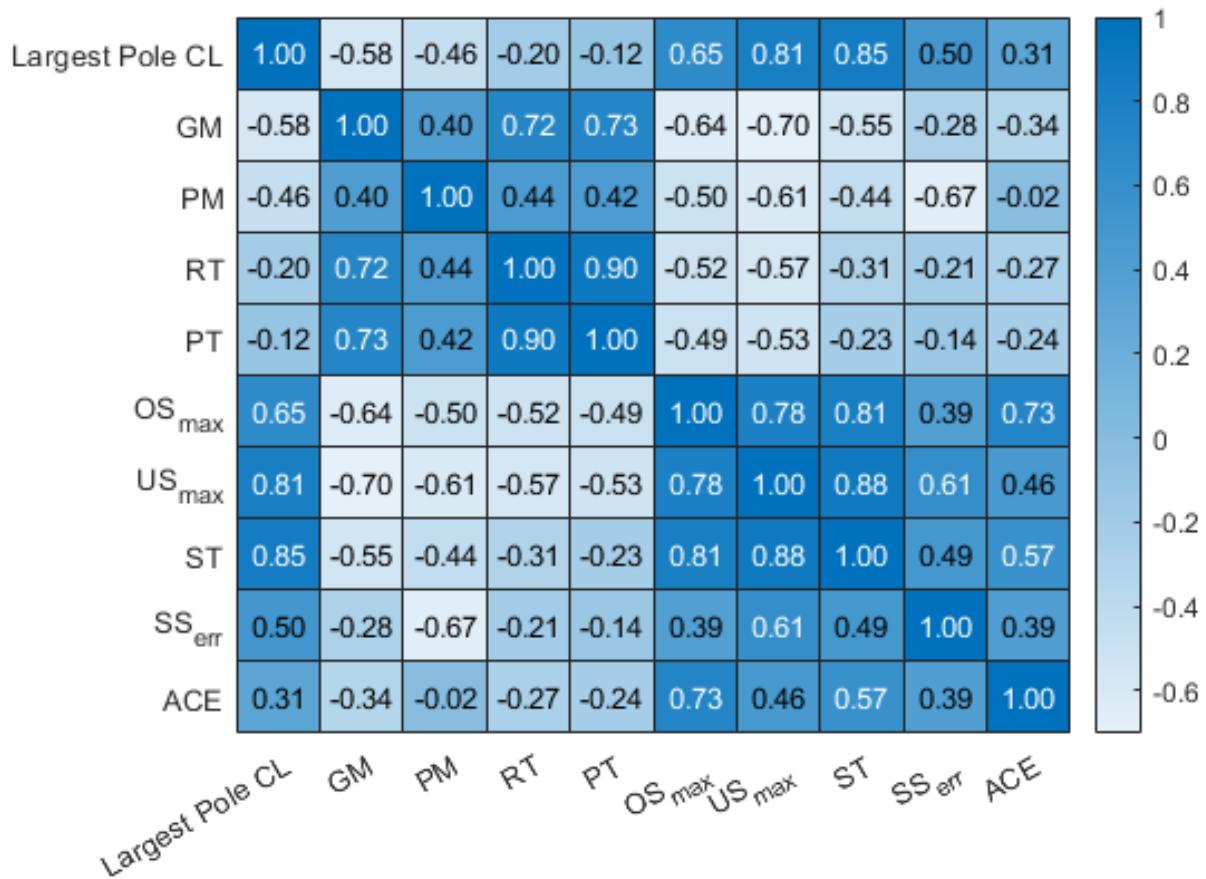


Figure 13: Heat Map Representing Performance Criteria Correlation for Full Factorial Sampling Plan

## D Scatter Plot and Sampling Post Optimisation (Sustainability Analysis)

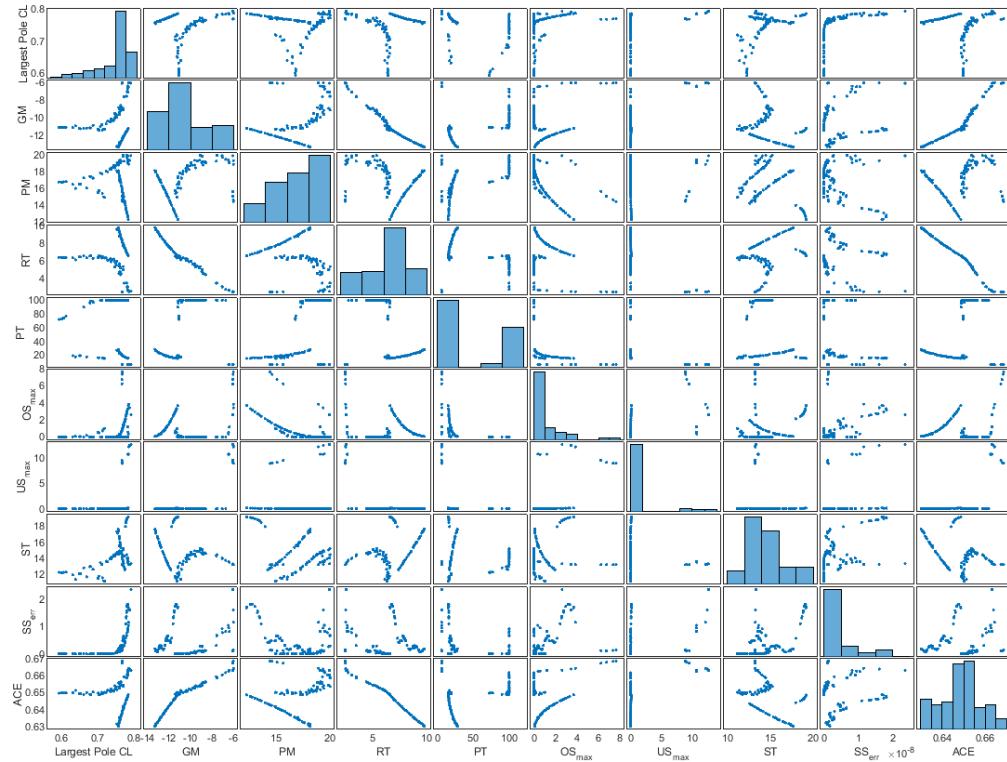


Figure 14: Scatter Plots (gplotmatrix with Reduced Aggregate Control Energy

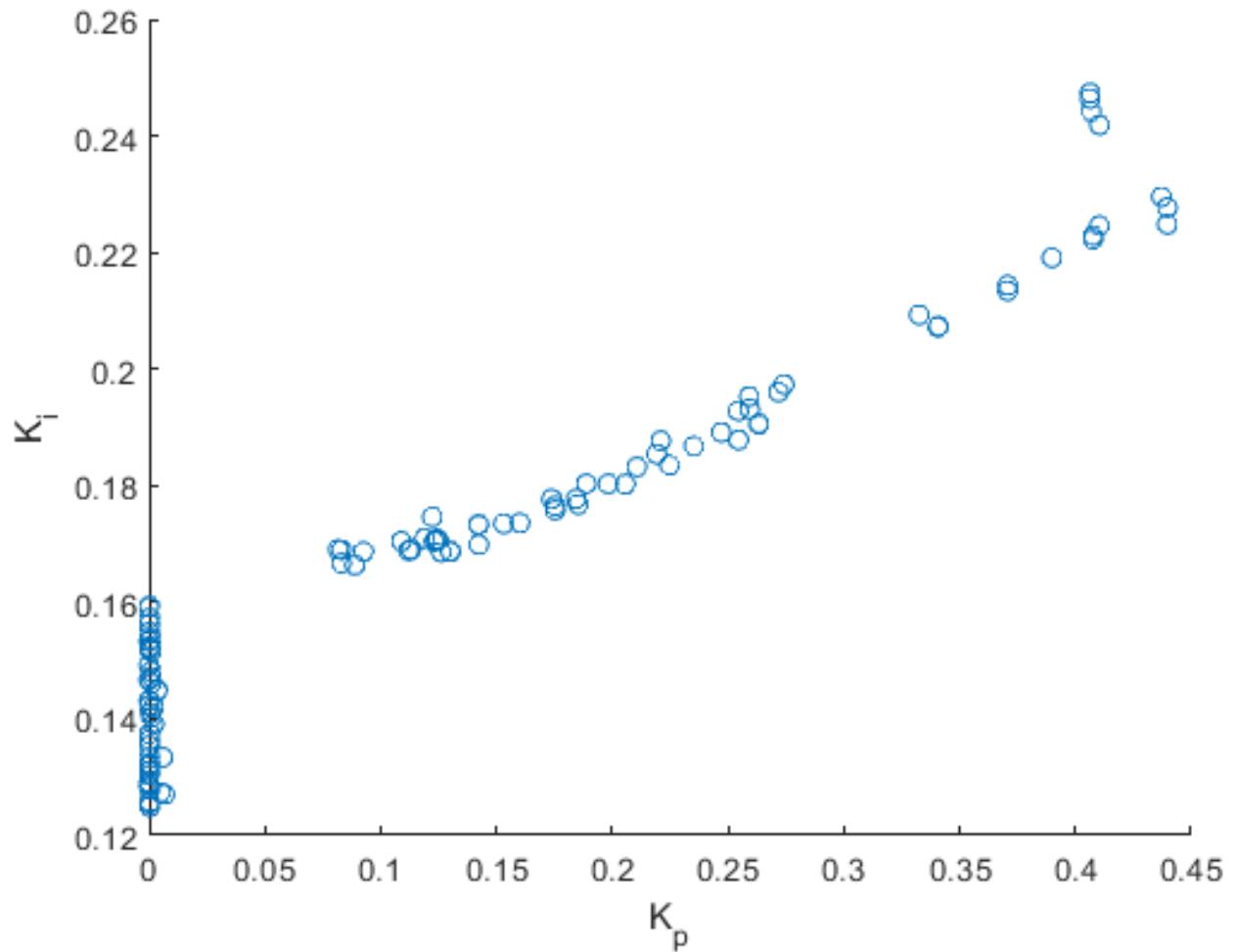


Figure 15: Sampling Update with Reduced Aggregate Control Effort Energy

## E MATLAB Code

---

```
1 clear all;
2 clc;
3
4 %% Sampling Plans
5
6 %Initializing q for the sampling plans
7 q = [10, 10];
8
9 % Full Factorial Sampling Plan
10 fullFactorial_sp = fullfactorial(q, 1);
11 fullFactorial_sp = 10*fullFactorial_sp + eps;
12
13 % RLH Sampling Plan
14 rlh_sp = rlh(100, 2, 5);
15 rlh_sp = 10*rlh_sp + eps;
16
17 % Sobol Sampling Plan
18 P = sobolset(2);
19 sobol_sp = net(P,100);
20 sobol_sp = 10*sobol_sp + eps;
21
22 % Scatter plots of sampling plans
23 figure;
24 scatter(fullFactorial_sp(:, 1), fullFactorial_sp(:, 2))
25 xlabel('K_p')
26 ylabel('K_i')
27 % title('Full Factorial Sampling')
28
29 figure;
30 scatter(rlh_sp(:, 1), rlh_sp(:, 2))
31 xlabel('K_p')
32 ylabel('K_i')
```

```

33 % title('RLH Sampling ')
34
35 figure;
36 scatter(sobol_sp(:, 1), sobol_sp(:, 2))
37 xlabel('K_p')
38 ylabel('K_i')
39 % title('Sobol Sampling')
40
41 % Assessing sampling plans using mmphi metric
42 fullFactorial_metric = mmphi(fullFactorial_sp, 5, 2); % Full
    Factorial
43 rlh_metric = mmphi(rlh_sp, 5, 2); % RLH
44 sobol_metric = mmphi(sobol_sp, 5, 2); % Sobol
45
46 %% Knowledge Discovery
47 % Evaluating full factorial design
48 Z = evaluateControlSystem(fullFactorial_sp);
49 % Y = [fullFactorial_sp, Z];
50
51 % Plotting the correlation of performance criteria and design
    variables using gplotmatrix
52 figure;
53 labels = ["Largest Pole CL", "GM", "PM", "RT", "PT", "OS_{max}", "
    US_{max}", "ST", "SS_{err}", "ACE"];
54 gplotmatrix(Z, [], [], [], [], [], 'on', [], labels(1:10),labels
    (1:10));
55
56 % Plotting the correlation of performance criteria and design
    variables using parallelplot
57 figure;
58 parallelplot(Z);
59
60 %% Heat Map
61 % labels for the axes

```

```

62 labels = ["Largest Pole CL", "GM", "PM", "RT", "PT", "OS_{max}", "US_{max}", "ST", "SS_{err}","ACE"];
63
64 % Identify rows that have inf values
65 hasInf = any(isinf(Z), 2);
66
67 % Remove rows that contain any Inf values
68 Z_inf_removed = Z(~hasInf, :);
69
70 % Compute the correlation matrix
71 corrMatrix = corr(Z_inf_removed);
72
73 % Create the heatmap
74 figure;
75 h = heatmap(corrMatrix);
76
77 % Set the labels for the axes
78 h.XDisplayLabels = labels;
79 h.YDisplayLabels = labels;
80
81 % Display the correlation values inside the heatmap cells
82 h.CellLabelFormat = '%.2f';
83
84 %% Optimization Process
85 % Applying post-processing conditions of Gain Margin and Phase
Margin before the optimization process
86 Z_eval = optimizeControlSystem(fullFactorial_sp);
87
88 % Initializing an empty vector for hypervolume indicator
89 HV_values = [];
90 % Reference points of hypervolume indicator
91 Reference_values = [max(Z_eval)];
92
93 % Loop for NSGA-II optimizer for 250 iterations

```

```

94 parent = fullFactorial_sp;
95 for i = 1:125
96     disp('iteration: ')
97     disp(i)
98 % Priorities: Hard_constraint =3; High =2; Moderate = 1; Low = 0;
99 % Progressive use of preferences over iterations
100 if i <= 50
101     Goals = [0.8 -inf -inf -inf -inf -inf -inf -inf -inf];
102     Priorities = [1 0 0 0 0 0 0 0 0];
103 elseif i > 50 && i <= 75
104     Goals = [0.8 -6 20 -inf -inf -inf -inf -inf 0.67];
105     Priorities = [2 1 1 0 0 0 0 0 1];
106 elseif i > 75 && i <=100
107 % Applying all the goals and priorities as per cheif engineer's
108     % preferences
109     Goals = [0.8 -6 20 2 10 10 8 20 1 0.67];
110     Priorities = [3 2 2 1 0 1 0 0 1 2];
111
112 else
113     Goals = [0.8 -6 20 2 10 10 8 20 1 0.67];
114     Priorities = [3 2 2 1 0 1 0 0 1 1];
115 end
116 % Incorporating rank preferability
117 [ranking,ClassV] = rank_prf(Z_eval,Goals, Priorities);
118
119 % crowd sorting
120 crowd = crowding(Z_eval,ranking);
121 ranking = max(ranking) - ranking;
122 % Binary tournament selection
123 binary = btwr([ranking,crowd],100);
124 % setting bounds for the cross and mutation
125 bounds = [0 0; 1 1];
126 % CrossOver and Mutation variation

```

```

127
128 cross = sbx(parent(binary ,:) ,bounds); % CrossOver
129
130 % Post Variation Children
131 child = polymut(cross ,bounds); % Mutation
132
133 % Concatinating Parent with child
134 P_C = [parent;child];
135
136 % Evaluating the concatenated Parent and child
137 Z_eval_PC = optimizeControlSystem(P_C);
138
139 % rank preferability for the concatenated Parent and child
140 [ranking_PC ,Class] = rank_prf(Z_eval_PC ,Goals , Priorities);
141
142 % crowd sorting for the concatenated Parent and child
143 crowd_PC = crowding(Z_eval_PC ,ranking_PC);
144
145 % NSGA-II selection for survival operator
146 reduceNSGA = reducerNSGA_II(P_C,ranking_PC,crowd_PC ,100);
147
148 % reduced designs of two populations
149 reduced_design = P_C(reduceNSGA ,:);
150
151 % Evaluating the reduced designs of two populations
152 Z_eval = optimizeControlSystem(reduced_design);
153
154 % ..... Hypervolume ..... %
155 % Indicator
156 HV = Hypervolume_MEX(Z_eval ,Reference_values);
157 % Vector
158 HV_values = [HV_values; HV];
159 % ..... %
160

```

```

161 % Updating sampling plan for next iteration
162 parent = reduced_design;
163
164 end
165
166 %% Plotting Optimised Results
167 figure;
168 scatter(parent(:, 1), parent(:, 2))
169 xlabel('K_p')
170 ylabel('K_i')
171 % title('Full Factorial Sampling After Optimisation')
172
173 % Plotting optimised correlation of performance criteria and design
174 % variables gplotmatrix)
174 figure;
175 labels = ["Largest Pole CL", "GM", "PM", "RT", "PT", "OS_{max}", " "
176 US_{max}", "ST", "SS_{err}","ACE"];
176 gplotmatrix(Z_eval, [], [], [], [], [], 'on', [], labels(1:10),
177 labels(1:10));
177
178 % Plotting the optimized correlation of design variables and
179 % performance criteria using gplotmatrix
179 figure;
180 parallelplot(Z_eval);
181
182 %% Plotting Hypervolume Result
183 figure;
184 plot(HV_values);
185 xlabel('Iteration');
186 ylabel('Hypervolume');
187
188 %% Functions
189 function Z = optimizeControlSystem(x)
190 Z = evaluateControlSystem(x);

```

```
191 % Gain Margin adjustment  
192 Z(:, 2) = -(20 * log10(Z(:, 2)));  
193 % Phase Margin adjustment  
194 Z(:, 3) = abs(Z(:, 3) - 50);  
195 end
```

---