

# Many-Objective Robust Gait Optimization for a 25-DOF NAO Robot Using NSGA-III

Pushpendra Gupta<sup>†</sup>, Dilip Kumar Pratihar <sup>\*†</sup>, and Kalyanmoy Deb<sup>‡</sup>

<sup>†</sup>*Mechanical Engineering Department, Indian Institute of Technology Kharagpur, Kharagpur-721302, West Bengal, India*

*pushpendra050@iitkgp.ac.in, dkpra@mech.iitkgp.ac.in*

<sup>‡</sup>*Electrical and Computer Engineering, Michigan State University, East Lansing, Michigan-48824, USA*  
*kdeb@egr.msu.edu*

## COIN Report 2025007

**Abstract** · Researchers increasingly employ evolutionary algorithms to tackle complex robotics problems due to their ability to handle nonlinear dynamics. Robust solutions that remain stable despite variations in design parameters are essential for decision-makers. This study investigates Type I and Type II robustness approaches within constrained many-objective optimization (MaOO) frameworks, which are rarely explored in humanoid robotics. It focuses on optimizing the gait cycle of a 25 DOF NAO humanoid robot during single- and double-support phases. The NSGA-III algorithm is employed to address four conflicting objectives, such as minimizing power consumption, maximizing stability, minimizing torque fluctuations, and reducing gait cycle time. A detailed comparative analysis highlights the superiority of Type II robustness, offering better-distributed and more convergent solutions in real-world scenarios with several constraints and variables. Furthermore, the study examines the influence of gait parameters on objectives, enhancing the understanding of humanoid robot dynamics and presenting a robust methodology for similar complex challenges.

**Keywords:** Many-objective evolutionary algorithm, Robust optimization, NSGA-III, Efficient Pareto-optimal front, NAO humanoid robot

## 1. Introduction

Humanoid robots have garnered significant attention in the research community due to their remarkable similarity to human movement patterns, offering the potential for versatile assistance in everyday environments [1]. Bipedal locomotion poses complex challenges due to its nonlinear dynamics and the coordination of numerous degrees of freedom, making traditional optimization methods inadequate for achieving stable and efficient gaits. The evolutionary multi-objective optimization (EMO) algorithms have emerged as powerful tools for addressing these challenges in humanoid robots and other robotic systems. EMO algorithms enable humanoid robots to achieve efficient movement, characterized by low power consumption [2,3], improved stability [2,3], better trajectory optimization [4–6], path planning [7,8], and gait planning [2,9,10] on uneven terrains, thereby enhancing overall performance. While existing research has made significant progress in multi-objective optimization for humanoid locomotion, most of the studies focus on optimizing a limited set of two to three objectives, such as power [3,5,11–17] and stability [3,4,14,17,18], speed [9,18], motor cost [18], minimum torque changes [11], and jerk minimization [19] highlighting the necessity for a comprehensive optimization approach that simultaneously considers more than three objectives. Since bipedal locomotion inherently involves more than three competing objectives, it naturally falls into many-objective optimization problems (MaOPs). These MaOPs introduce new challenges in algorithm design, visualization, and practical implementation. Despite recent advances in humanoid robot locomotion, the application of evolutionary many-objective optimization (EMaO) algorithms in this field remains largely unexplored.

---

\*Corresponding Author

This research addresses this gap by solving an MaOP for the gait cycle of a 25 DOF NAO humanoid robot using EMO algorithms.

Emerging trends in EMO and EMO algorithms have revolutionized complex problem-solving in the field of robotics. These algorithms identify multiple non-dominated (ND) solutions, forming a Pareto-front (PF) that balances conflicting objectives and offers decision-makers (DMs) diverse, preference-driven options. However, when implementing these solutions to optimize robotics problems, challenges arise. Small variations in design parameters like electrical component tolerances, external forces, and motor control uncertainties can lead to unexpected results. DMs now seek robust solution sets that remain stable even when minor changes occur in the decision variables. Researchers focusing on these algorithms have mainly explored two types of robustness approaches, known as Type I and Type II, to tackle the problems with multi- and many-objectives [20]. Robustness is essential in these algorithms, as DMs require solutions that perform reliably despite uncertainties and parameter variations, beyond just achieving an optimal PF. However, the use of these approaches in real-world situations, particularly those with many objectives and several constraints, has been less explored. This study integrates the Types I and II robustness methodologies into MaOPs and compares them to advancing humanoid robot locomotion. This study prioritizes the robustness to ensure reliable solutions to handle real-world uncertainties.

In addressing the challenges of many-objective robust optimization (MaORO) for humanoid locomotion, the current study focuses on the 25-DOF NAO humanoid robot, which has garnered significant attention from researchers [16, 21–26] due to its compact size, affordability, and software development capabilities. The present investigation tackles a constrained MaOP that simultaneously addresses four critical objectives: minimizing power consumption, maximizing stability, minimizing torque fluctuations, and minimizing gait cycle time using the non-dominated sorting genetic algorithm III (NSGA-III) [27–29]. The optimization process separately analyzes the single support phase (SSP) and double support phase (DSP), incorporating arm movements to assess their impact on dynamic stability and achieve a balanced gait. The aim is to prioritize robustness, evaluating Type I and Type II robustness approaches to ensure that the solutions remain effective under parameter variations and uncertainties. This study demonstrates NSGA-III’s efficacy in solving practical MaOPs, while equipping DMs with insights into solution robustness for reliable performance. Furthermore, the selected optimization problem serves as a challenging benchmark, similar to other complex engineering problems. As a result, the proposed methodology can be readily applied to practical scenarios with comparable or even greater complexity. This study not only contributes to advances in EMO techniques, but also provides valuable insight into the complex relationships between gait parameters and objectives.

The rest of the article is organized in the following manner: Section 2 presents a comprehensive review of related work in bipedal locomotion, MaOO, and robustness in humanoid robotics. The mathematical formulation of the gait generation problem is presented in Section 3, while the MaOP for both phases is detailed in Section 4 respectively. Section 5 elaborates on the optimization approaches, and Section 6 discusses the methodology employed to analyze Type I and II robustness. The results are discussed in Section 7. Lastly, Section 8 provides concluding remarks.

## 2. Literature Survey

This section reviews studies on evolutionary computation and robustness in humanoid robotics. It concludes by identifying research gaps and outlining the contributions of the present study.

### 2.1. Evolutionary Computation-based Studies

The application of EMO algorithms in bipedal locomotion has shown remarkable progress over the past decades, particularly in addressing the challenges of energy efficiency and stability. Several researchers have highlighted the utility of EMO algorithms in discovering energy-efficient trajectories and stable bipedal gait cycles [2–4, 14, 17, 30–32]. Gupta et al. [17] applied EMO algorithm to optimize the gait cycle of a 25-DOF NAO humanoid robot, focusing on two conflicting objectives: power consumption and dynamic stability. They modified the NSGA-II employing angle- and utility-based methods to identify "knee" solutions within the Pareto front, aiding decision-making by reducing the

trade-off space to the most efficient solutions. Rajendra and Pratihari [2] addressed the challenge of planning the leg movement of a biped robot on staircases while balancing the goals of reducing power consumption and increasing the dynamic stability margin. Raj et al. [3] explored the trade-off between dynamic stability and energy consumption to enhance the NAO robot’s gait cycle. Uno et al. [11] observed minimal torque changes along with low energy consumption and found that, while energy-efficient walking cycles closely resemble human motion, those with minimal torque changes exhibited enhanced stability. Sanprasit [19] compared the effectiveness of four EMO algorithms in achieving the desired trade-off between stability and jerk minimization for the robot’s walking path design. EMO algorithms were employed in an optimal fuzzy tracking control approach to achieve stable walking in bipedal robots [4]. Leng et al. [18] addressed the intricate balance between speed, stability, and motor cost, revealing inherent conflicts among these objectives. Similarly, Kulvanit et al. [9] applied an EMO algorithm to accelerate the locomotion of bipedal robots, while Muni et al. [31] utilized the EMO algorithm to refine the NAO robot’s navigational capabilities. The contributions of Vikas et al. [16] and Maafi et al. [5] have been pivotal in advancing path planning and trajectory tracking, focusing on coordination, obstacle avoidance, and energy efficiency. Liu et al. [33] contributed to the development of multi-locomotion robots using an EMO algorithm. Researchers have utilized a diverse range of EMO algorithms to address the dynamic performance in robotics.

It is clear from previous discussions that past research had primarily focused on optimizing two to three objectives at the same time. Only a few studies have attempted to optimize more than three objectives simultaneously in robotics applications. Russo et al. [34] considered four objectives to determine the ideal geometrical proportions for the links of a 3-UPR parallel mechanism. Kiyokawa et al. [35] developed a many-objective optimized semi-automated robotic disassembly sequences algorithm based on NSGA-III for optimizing disassembly operations. They designed four objective functions related to difficulty, efficiency, prioritization, and allocability, each aiming to minimize specific aspects of the disassembly process. The research has shown that EMaO algorithms are effective in solving complex MaOP in robotics. They have great potential for wider use in humanoid robotics. However, their use in bipedal locomotion remains limited despite the remarkable versatility and effectiveness of these approaches.

## 2.2. Robustness-based Studies

There is a growing interest among researchers in contributing to the development of various robotic systems with the aim of enhancing their robustness to handle real-world uncertainties. This effort is part of a broader research initiative dedicated to creating autonomous systems that are capable of navigating and performing tasks amidst the unpredictable and complex conditions of the real world. Recent advancements in robotics research have led to significant progress in areas such as robust humanoid robot walking [36–45], robustness in search and rescue operations [46, 47], enhancing the robustness of human-robot collaboration [48, 49], robust trajectory optimization [41, 50–54], robust localization and mapping [55], robust analysis for the mechanism of a virtual monopedal [56], robust control for a robot [57–59], robust solutions for robotic task from optimization algorithms [60], and robust task priorities [61]. These studies have focused on enhancing the robustness of robots to perform reliably under various uncertainties, including sudden disturbances [40, 59], uncertain and perceptually-degraded environments [47], motion classification noises [49], variations in floor properties [36], complex and uneven terrain [37, 38, 43, 48, 51], modeling errors [42, 50, 61], indoor and outdoor industrial environments [55], real-world imperfections on performance [56], uncertain dynamic environment [53, 57], model discrepancies [42, 60], and inaccurate modeling of the robot-environment interaction [52, 58].

Researchers have developed diverse approaches to enhance robustness in bipedal locomotion. Ito and Hasegawa [36] addressed the challenge of optimizing bipedal gait without relying on sensors, using simulated annealing to account for random variations in floor properties and enabling stable walking even with delayed actuator or sensor responses. Collette et al. [37] proposed an optimization-based framework for humanoid posture control that considers both grasp and friction, handling external disturbances and non-coplanar contacts to improve stability in challenging scenarios like climbing. Yeganegi et al. [41, 42] addressed the challenges of model predictive control under uncertainties, propos-

ing a two-level model predictive control framework with Bayesian optimization to determine optimal cost weights for robust performance. Furthermore, [43] highlighted the importance of accurate state estimation, particularly velocity and height maps, in ensuring robust control of humanoid robots through learning-based policies. These studies collectively demonstrate the significant progress in enhancing the robustness of humanoid robots walking under various uncertainties. Gasparri et al. [48] addressed the robustness of robot compliance in scenarios where interaction with the environment or humans is unpredictable. They utilized robust optimization to determine the optimal level of compliance for robots performing tasks under uncertain conditions.

The research on robust trajectory optimization has also seen significant advancements. Luo and Hauser [50] tackled the challenge of robot trajectory optimization in the face of modeling errors, introducing a method that combines robust optimization with parameter learning to manage uncertainties in friction estimates, contact points, and actuator noise. Their robust time-scaling approach calculates a dynamically-feasible, minimum-cost trajectory under frictional contact, ensuring efficient task execution even with large modeling errors. Building upon this, Luke Drnach and Ye Zhao [51] presented a risk-sensitive method for optimizing contact-implicit trajectories in unpredictable terrains. They employed probability distributions to model terrain uncertainties and developed an Expected Residual Minimization (ERM) cost approach. This led to the creation of trajectories that are resilient to changes in the terrain, encouraging actions such as short and fast sliding movements or higher steps to maintain distance from the uncertain terrain. The effectiveness of the ERM strategy is shown to surpass that of the robust worst-case solution, as it yields control trajectories with inherent robustness. Gkikakis and Featherstone [56] introduced a robust analysis method for optimizing the mechanisms and behaviors of a virtual monopodal robot. Their approach accounts for real-world imperfections by considering realistic models and modeling uncertainties as statistical distributions. This prevents over-optimization and leads to robot designs that perform consistently and meet design expectations despite real-world variations.

Jorgensen et al. [60] explored the use of optimization algorithms to find robust solutions for robotic tasks, emphasizing the importance of parameterization and objective score selection. They highlighted the RBFopt algorithm’s efficiency and its ability to focus the search in regions that generally yield robust solutions. On the other hand, Char et al. [61] concentrated on optimizing task priorities for whole-body control of humanoid robots through domain randomization. This technique ensures that optimized solutions are viable in diverse working conditions, facilitating a successful transfer from simulation to real-world application. These studies highlight the critical role of robustness in achieving reliable humanoid robot performance under real-world conditions.

### 2.3. Research Gaps and Contributions

The literature review reveals several critical gaps in current approaches to bipedal locomotion optimization. While DSP [2, 3, 62] is integral to maintaining steady locomotion, particularly at moderate speeds [63], it has often been neglected in favor of SSP [64–68], due to the transient nature of DSP within the gait cycle. However, this research delves into both SSP and DSP of bipedal locomotion, due to the different power and stability calculations inherent to each phase. Traditionally, DSP modeled as a simple inverted pendulum [62]; such simplifications overlook the intricate kinematic structure of humanoid robots. By analyzing DSP as two separate SSPs, this study offers a more detailed analysis of the impact of multi-dynamic bodies on power consumption and stability, thereby providing a more holistic understanding of each phase in the NAO robot’s locomotion. While previous research had investigated diverse approaches to optimize bipedal robot gait patterns, the inclusion of lateral movements in these studies has remained limited [2, 13, 68]. In contrast, this study explicitly incorporates lateral movements into the gait planning process. Previous studies on NAO robots had predominantly focused on maintaining a constant hip height [3]. However, this research expands the scope by examining hip height variations across three dimensions. While several studies had focused on maintaining a constant hip height [3, 69] and constant velocity [69], only a few had investigated how changes in forward velocity impact energy consumption [70]. In contrast, this study specifically investigates the impact of transition velocities on power and stability, providing a more comprehensive understanding of their influences on bipedal locomotion. Previous research had frequently neglected the role of arm

movements in gait motion, with only a few studies [12, 40] examining arm movement during gait optimization across various walking phases. The comprehensive impact of arm movement on stability remains underexplored. This study, however, addresses this oversight by examining the influence of arm movements on dynamic stability. While numerous studies had concentrated on humanoid robots with 7 DOF [65, 67, 68, 71], this study expands the scope by considering a humanoid robot with 25 DOF, providing a more detailed analysis of the robot’s dynamic behavior.

It is evident from the earlier discussion that there have also been limited studies on real-world problems to demonstrate the robustness of MaOP in the gait cycle of humanoid robots. Therefore, to bridge this research gap, a real-world problem focusing on gait generation for bipedal locomotion in NAO humanoid robots has been considered. Studies of robust solutions that address highly constrained MaOPs of humanoid locomotion are scarce. Moreover, building upon the foundational work of Gupta et al. [21], which focused on power optimization for the NAO robot, this study adopts a holistic approach by simultaneously optimizing many conflicting objectives. This research paves the way for more robust and practical enhancements to NAO robot performance by optimizing these objectives simultaneously, rather than treating them as secondary constraints.

### 3. Mathematical Formulation of Gait Generation Problem

The bipedal locomotion in humanoid robots involves synchronized leg movements, including SSP and DSP [21] (refer to Figs. 1 and 3). Both of these phases are mathematically formulated in this section. The NAO robot is equipped with 25 DOF, and for the purposes of this study, all 25 associated lumped masses, as specified on the official website [72], have been taken into account for the analysis of both the SSP and DSP. The lumped masses have a distribution of six per leg and five per arm [73]. The last three lumped masses are allocated to the torso, neck, and head parts. It is postulated that the friction during both the SSP and DSP is adequate to avert any slippage. The Denavit-Hartenberg (DH) parameters [74] are utilized to assign coordinate frames to the robot’s joints, facilitating the study of its motion. Prior research on the NAO robot has investigated diverse methodologies [75–80] to solve the Inverse Kinematics (IK) challenge. The present study employed IK solutions developed by Nikolaos et al. [75]. The field of inverse dynamics is concerned with determining the joint torques necessary to produce a desired motion trajectory for the known joint angles. Researchers had applied various methodologies to study the inverse dynamics [77, 81–87]. The Lagrange-Euler formulation [83, 84, 88] is adopted to precisely compute the torques required at each joint ( $\tau_i$ ) [89], using the equation  $\tau_i = \sum_{c=1}^n \mathcal{D}_{ic} \ddot{\phi}_c + \sum_{c=1}^n \sum_{d=1}^n h_{icd} \dot{\phi}_c \dot{\phi}_d + \mathcal{C}_i$  for the NAO humanoid robot, to facilitate the desired motion. Here,  $\mathcal{D}_{ic}$  denotes the inertia matrix,  $h_{icd}$  the coriolis forces, and  $\mathcal{C}_i$  the gravitational forces.  $\dot{\phi}_i$  and  $\ddot{\phi}_i$  represent the angular velocity and acceleration of the joint, respectively. The terms  $\dot{\phi}_i$  and  $\ddot{\phi}_i$  represent the angular velocity and acceleration at the  $i$ th joint, respectively.

The concept of the Zero Moment Point (ZMP) [90] is utilized to ensure dynamic stability as the robot moves. The movements in the forward, lateral, and vertical directions are analyzed along the  $x$ ,  $y$ , and  $z$  axes, respectively. The ZMP must consistently be located within the boundary of the support polygon to maintain the NAO robot’s dynamic stability. The ZMP is defined as the particular ground point about which the sum of the moments due to forces becomes zero. The expressions for ZMP on the  $x$  and  $y$  axes are given in Eqs. (1) and (2). Each symbol is used in accordance with its standard definition [88].

$$X_{ZMP} = \frac{\sum_{i=1}^{i=n} (I_i \dot{\omega}_i + m_i x_i (\ddot{z}_i - g) - m_i \ddot{x}_i z_i)}{\sum_{i=1}^n m_i (\ddot{z}_i - g)} \quad (1)$$

$$Y_{ZMP} = \frac{\sum_{i=1}^{i=n} (I_i \dot{\omega}_i + m_i y_i (\ddot{z}_i - g) - m_i \ddot{y}_i z_i)}{\sum_{i=1}^n m_i (\ddot{z}_i - g)} \quad (2)$$

### 3.1. Single Support Phase (SSP) Scenario

Figs. 1 and 2(a) depict the actual and kinematic representations of the NAO humanoid robot [91] in the SSP. The stride length is determined by the initial and final positions of the swinging ankle symbolized by  $x_i$  and  $x_f$ . The stride length being the distance between these two points set at 0.12 meters. The maximum swing height, represented as  $S_h^{max}$ , is achieved at the midpoint of the stride. The step length is denoted as  $S_L$ , while the hip height is defined by the parameter  $Z_H$ . For the analysis of SSP, the robot is conceptualized as a serial manipulator, initiating from the left foot and sequentially integrating the seven distinct segments of the robot, ending at the right foot. The determination of the ground reaction force's magnitude and position is an important factor in bipedal locomotion analysis.

The robot's combined Center of Mass (CoM) is computed using the formula  $C = \sum_{i=1}^{i=n} m_i p_i / \sum_{i=1}^{i=n} m_i$ . where  $m_i$  represents the mass and  $p_i \in \mathbb{R}^{3 \times 1}$  signifies the position of the CoM for the  $i$ -th lumped mass. The rate of change of the CoM,  $\dot{C}$ , is obtained by differentiating  $C$ , resulting in  $\dot{C} = \sum_{i=1}^{i=n} m_i \dot{p}_i / \sum_{i=1}^{i=n} m_i$ . The total momentum of the robot, denoted as  $P'$ , is equal to  $\sum_{i=1}^{i=n} m_i \dot{p}_i$ . The translational motion of the NAO robot is characterized by the equation  $\dot{P}'$ , which is defined as  $\dot{P}' = \sum_{i=1}^{i=n} m_i \ddot{p}_i = \sum_{i=1}^{i=n} f_i$ . Here,  $f_i$  represents the three-dimensional force vector acting on the  $i$ -th lumped mass. The total external force exerted on the NAO robot, symbolized as  $F$ , is the sum of all individual forces ( $\sum_{i=1}^{i=n} f_i$ ). This total force  $F$  comprises two components: the gravitational force ( $Mg$ ) and ground reaction force ( $f$ ), where  $M$  is the total mass in kilograms and  $f \in \mathbb{R}^{3 \times 1}$  has components along the  $x$ ,  $y$ , and  $z$  axes. The force's magnitude is indicated by  $R$ , and its direction is illustrated in Fig. 2 (a). The point of application of this force determines the ZMP, depicted in Fig. 2 (b). The robot's foot dimensions, which provide context for its shape and size, are presented in Fig. 2 (c). The ZMP's minimum and maximum values are displayed in Fig. 2 (d).

The Dynamic Balance Margin (DBM) [92], a positive metric, measures the minimum distance between the ZMP and the boundary of the ground contact polygon. The computation of DBM, which depends on the foot position and ZMP, is detailed in Eqs. (3) and (4) for the  $x$  and  $y$  axes, correspondingly.

$$X_{DBM} = \begin{cases} (X_i + 0.1) - X_{ZMP}, & \text{if } X_{ZMP} \in [X_i, X_i + 0.1] \\ X_{ZMP} - (X_i - 0.05), & \text{if } X_{ZMP} \in [X_i - 0.05, X_i] \\ 0, & \text{otherwise} \end{cases} \quad (3)$$

$$Y_{DBM} = \begin{cases} (Y_i + 0.04) - Y_{ZMP}, & \text{if } Y_{ZMP} \in [Y_i, Y_i + 0.04] \\ Y_{ZMP} - (Y_i - 0.04), & \text{if } Y_{ZMP} \in [Y_i - 0.04, Y_i] \\ 0, & \text{otherwise} \end{cases} \quad (4)$$

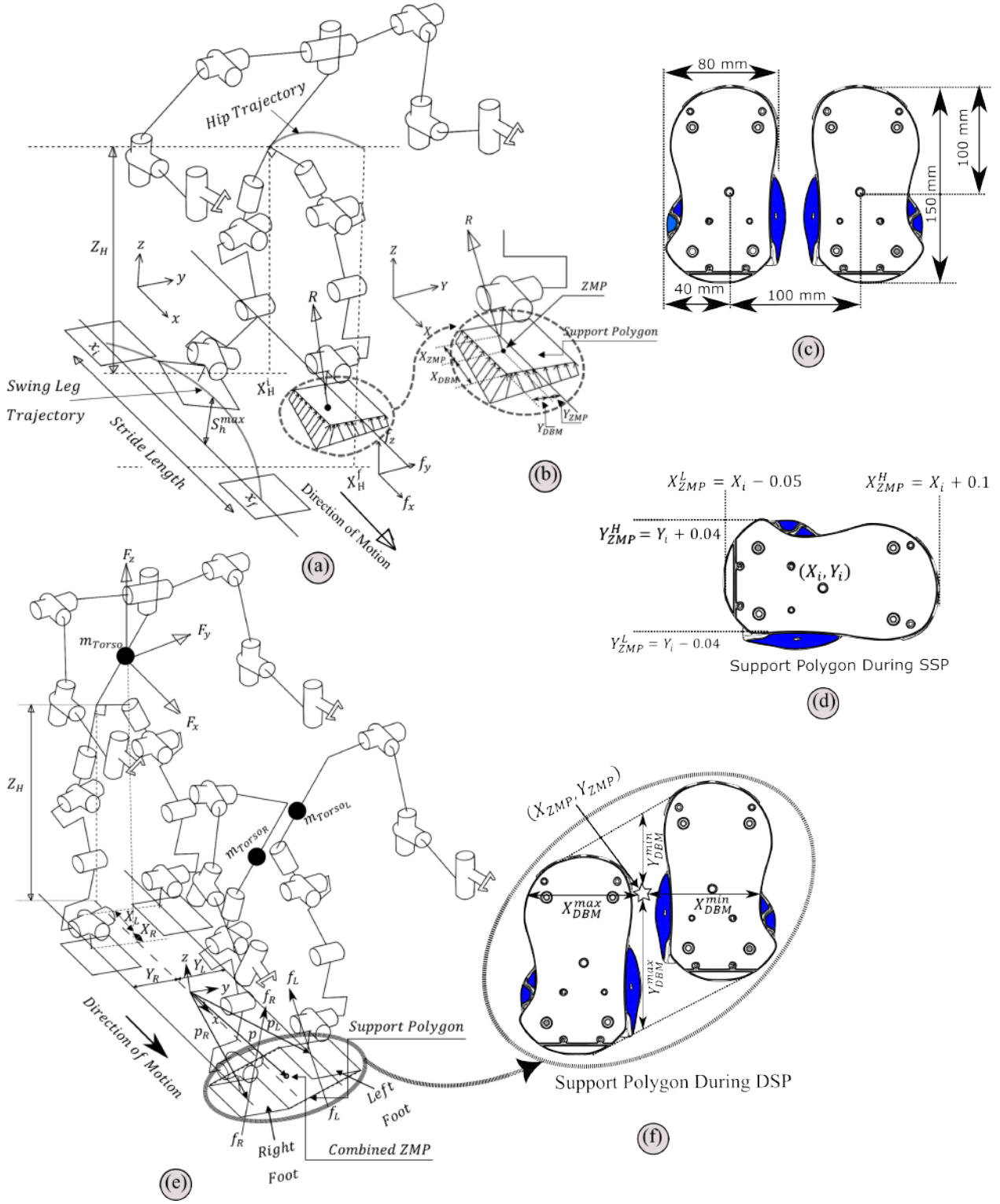
The trajectories of the hip, swinging leg, and arms are essential during the robot's movement. These trajectories are formulated through a series of cubic polynomial equations that outline the intended motion patterns for the movements of the ankle, hip, and arms. The cubic polynomial equations and their associated boundary conditions are detailed in Table 1 (Sr. No. 1–7) during SSP. The boundary conditions are used in determining the cubic equations' unknown coefficients, which are crucial for the precise planning and regulation of the robot's movements in the ankle, hip, and arms during the SSP phase. Figure 2 (a) illustrates the symbols employed in Table 1. The IK process is then used to determine the joint angles.



**Fig. 1.** Single Support Phase

Table 1: Additional problem information

Sr. No.	Ankle, hip, and arm's trajectories	SSP Trajectory	
		SSP Trajectory	Boundary Conditions
1 <sup>c</sup>	$X_A(t) = a_0 + a_1t + a_2t^2 + a_3t^3$	$X_A(t_f) = x_f, \dot{X}_A(t_f) = 0, \ddot{X}_A(t_f) = 0$	
2 <sup>c</sup>	$Z_A(X_A) = b_0 + b_1X_A + b_2X_A^2 + b_3X_A^3$	$Z_A(x_i) = 0, Z_A(x_i + S_L) = 0, Z_A(x_m) = S_h^{max}, \dot{Z}_A(x_m) = 0$	
3 <sup>c</sup>	$X_H(t) = x_0 + x_1t + x_2t^2 + x_3t^3$	$X_H(t_f) = x_i + x_f/4, \dot{X}_H(t_f) = x_i + 3x_f/4, \ddot{X}_H(t_i) = V_s^X, \dot{X}_H(t_f) = V_e^X$	
4 <sup>c</sup>	$Y_H(t) = y_0 + y_1t + y_2t^2 + y_3t^3$	$Y_H(t_i) = 0.025 m, Y_H(t_f) = 0.025 m, \dot{Y}_H(t_i) = V_s^Y, \dot{Y}_H(t_f) = V_e^Y$	
5 <sup>c</sup>	$Z_H(t) = z_0 + z_1t + z_2t^2 + z_3t^3$	$Z_H(t_i) = h, Z_H(t_f) = h, \dot{Z}_H(t_i) = V_s^Z, \dot{Z}_H(t_f) = V_e^Z$	
6 <sup>j</sup>	$\phi_{SP}(t) = \alpha_0 + \alpha_1t + \alpha_2t^2 + \alpha_3t^3$	$\phi_{SP}(t_i) = q_i^{ShoulderPitch} rad, \phi_{SP}(t_f) = q_f^{ShoulderPitch} rad, \dot{\phi}_{SP}(t_i) = 0, \dot{\phi}_{SP}(t_f) = 0$	
7 <sup>j</sup>	$\phi_{ER}(t) = \beta_0 + \beta_1t + \beta_2t^2 + \beta_3t^3$	$\phi_{ER}(t_i) = q_i^{ElbowRoll} rad, \phi_{ER}(t_f) = q_f^{ElbowRoll} rad, \dot{\phi}_{ER}(t_i) = 0, \dot{\phi}_{ER}(t_f) = 0$	
DSP Trajectory			
8 <sup>c</sup>	$X_H(t) = x_0 + x_1t + x_2t^2 + x_3t^3$	$X_H(t_f) = 0.07 m, X_H(t_f) = 0.11 m, \dot{X}_H(t_i) = V_s^X, \dot{X}_H(t_f) = V_e^X$	
9 <sup>c</sup>	$Y_H(t) = y_0 + y_1t + y_2t^2 + y_3t^3$	$Y_H(t_f) = 0.025 m, Y_H(t_f) = -0.025 m, \dot{Y}_H(t_i) = V_s^Y, \dot{Y}_H(t_f) = V_e^Y$	
10 <sup>c</sup>	$Z_H(t) = z_0 + z_1t + z_2t^2 + z_3t^3$	$Z_H(t_i) = h, Z_H(t_f) = h, \dot{Z}_H(t_i) = V_s^Z, \dot{Z}_H(t_f) = V_e^Z$	
11 <sup>j</sup>	$\phi_{SP'}(t) = \gamma_0 + \gamma_1t + \gamma_2t^2 + \gamma_3t^3$	$\phi_{SP'}(t_i) = q_i^{ShoulderPitch} rad, \phi_{SP'}(t_f) = q_f^{ShoulderPitch} rad, \dot{\phi}_{SP'}(t_i) = 0, \dot{\phi}_{SP'}(t_f) = 0$	
12 <sup>j</sup>	$\phi_{ER'}(t) = \delta_0 + \delta_1t + \delta_2t^2 + \delta_3t^3$	$\phi_{ER'}(t_i) = q_i^{ElbowRoll} rad, \phi_{ER'}(t_f) = q_f^{ElbowRoll} rad, \dot{\phi}_{ER'}(t_i) = 0, \dot{\phi}_{ER'}(t_f) = 0$	
<p>Trajectories are planned in <sup>c</sup>cartesian space and <sup>j</sup>joint-space.</p> <p><math>[X_A(t), Y_H(t) \pm 0.05, Z_A(X_A)]^T</math>: Swing leg Ankle Trajectories, <math>+0.05</math>: Left Leg, <math>-0.05</math>: Right Leg, <math>[X_H(t), Y_H(t), Z_H(t)]^T</math>: Hip Trajectories <math>t_i</math> and <math>t_f</math> are the initial and final times, respectively, during one gait cycle. <math>S_L</math> is the Step length kept equal to <math>0.06 m</math>.</p> <p><math>h</math> represents the initial and final hip height. The support leg is fixed at <math>[x_i + x_f/2, 0.05, 0]^T</math> during the entire swing leg movement.</p> <p><math>V_s^X, V_s^Y</math> and <math>V_s^Z</math> are the starting velocities in <math>x, y</math>, and <math>z</math>-directions, respectively.</p> <p><math>V_e^X, V_e^Y</math> and <math>V_e^Z</math> are the end velocities in <math>x, y</math>, and <math>z</math>-directions, respectively.</p> <p>The hip traveled from <math>X_H(t_i)</math> to <math>X_H(t_f)</math>.</p> <p><math>\phi_{SP}</math> and <math>\phi_{ER}</math> stands for Shoulder Pitch and Elbow Roll trajectories in joint-space during SSP.</p>			



**Fig. 2.** (a) A line diagram illustrating the trajectory of the Hip and Swing leg and the ground reaction force  $R$  during the SSP. (b) ZMP and DBM Definitions (c) Dimensions of the NAO robot's foot. (d) ZMP bounds during SSP: upper and lower limits denoted by superscripts  $H$  and  $L$ , respectively.  $[X_i, Y_i]$  denotes the foot coordinates. (e) A schematic representation of the DSP, divided into two SSPs, showcasing the ground forces exerted during each SSP. The combined ZMP is depicted within the support polygon during DSP. (f) Minimum  $X_{DBM}$  and  $Y_{DBM}$  values based on coordinates from both feet and the ZMP.  $[X_l, Y_l]$  and  $[X_r, Y_r]$  represent the left and right foot coordinates, respectively.

The role of the humanoid robot's arms is crucial in preserving dynamic balance throughout its movement. Alterations in the configuration of the arms can significantly influence the robot's overall dynamic stability as it walks. Additionally, these changes can modify the torque demands on the arms. This study takes into account the effects of arm configuration variations on dynamic stability and torque requirements. This comprehensive approach ensures a thorough understanding of the influence that different arm configurations have on the robot's movement, as well as their subsequent effects on dynamic stability and torque demands. The Shoulder Roll angles are fixed at 0.26 rad for the left arm and  $-0.26$  rad for the right arm, while the Elbow Yaw angles are kept fixed at 1.5 rad for the left arm and  $-1.5$  rad for the right arm. The Wrist Yaw angles for both arms are maintained at zero. To assess the impact of the arms on dynamic stability, the trajectories for the Shoulder Pitch ( $\psi^{SP}$ ) and Elbow Roll ( $\psi^{ER}$ ) joints are examined. These trajectories are defined by cubic polynomials. Table 1 (Sr No. 6–7) details the four boundary conditions necessary to find the coefficients for the arm trajectories.

### 3.2. Double Support Phase (DSP) Scenario

Fig. 3 shows the NAO humanoid robot during the DSP, characterized by both legs being grounded and offering stability to the robot. In this stage, the right leg, which was previously in motion, touches down, allowing the robot's torso to shift its weight from the left to the right leg. Fig. 2(e) presents the kinematic model of the NAO humanoid robot throughout the DSP.

The analysis of the DSP was initially conceptualized by Rajendra and Pratihara [2] as a combination of two SSPs. Their study faced challenges in computing the combined ZMP for the entire humanoid robot under concurrent ground forces from both SSPs, which led to the introduction of the Virtual ZMP concept. This research incorporates the methodology proposed by Gupta et al. [21], effectively resolving these challenges.

In this approach, the DSP is treated as two distinct SSPs. Within the sagittal plane,  $X_L$  and  $X_R$  denote the distances from the vertical projection of the torso mass to the left and right foot, respectively. Correspondingly,  $Y_L$  and  $Y_R$  represent these distances within the lateral plane. The NAO robot is represented as a simplified model comprising 23 point masses, which includes six masses for each leg, five for each arm, and one mass for the torso that also encompasses the neck and head masses. The robot's movements and dynamic stability are considered in both the sagittal and lateral planes, with the robot achieving stability by distributing its weight across both legs during the DSP. The first SSP includes the left leg, left arm, and torso, while the second SSP comprises the right leg, right arm, and torso, allowing for an exhaustive analysis of the robot's dynamics and stability during the DSP.

The torso mass,  $m_{Torso}$ , is divided between the lateral and sagittal planes, apportioned as  $m_{TorsoL} = m_{Torso}Y_L/(Y_L + Y_R)$  for the left leg and  $m_{TorsoR} = m_{Torso}Y_R/(Y_L + Y_R)$  for the right leg. The external force vector  $F$  acting on the NAO robot is expressed as  $F = [F_x, F_y, F_z]^T$ , calculated similarly to the SSP. The force components in the  $x$ ,  $y$ , and  $z$  directions are depicted in Fig. 2 (e). The force  $f$ , ascertained during the SSP, is denoted by  $f_L$  and  $f_R$  for the left and right legs, respectively, with  $f_L$  and  $f_R$  having components  $[f_{Lx}, f_{Ly}, f_{Lz}]^T$  and  $[f_{Rx}, f_{Ry}, f_{Rz}]^T$ , respectively.

For ZMP calculations, where  $n$  is set to 12 (refer to Eqs. (1) and (2)), the value of  $m_{Torso}$  is adjusted based on the leg being analyzed  $m_{TorsoL}$  for the left leg and  $m_{TorsoR}$  for the right leg. The ZMP's position (point  $p$ ) for the entire model during DSP is determined by setting the  $x$  and  $y$  moment components about point  $p = [p_x, p_y, p_z]^T$  to zero [93], resulting in  $p_x = (p_{Lx}f_{Lz} + p_{Rx}f_{Rz})/(f_{Lz} + f_{Rz})$  and  $p_y = (p_{Ly}f_{Lz} + p_{Ry}f_{Rz})/(f_{Lz} + f_{Rz})$ , with  $p_L = [p_{Lx}, p_{Ly}, p_{Lz}]^T$  and  $p_R = [p_{Rx}, p_{Ry}, p_{Rz}]^T$  representing the positions of the ZMP. On flat ground, the  $p_z$  component of  $p$  is considered to be zero due to the absence of vertical displacement. The DBM during DSP is computed by determining the



**Fig. 3.** Double Support Phase

shortest distance between the ZMP and the boundary of the ground contact polygon in the  $x$  and  $y$  planes, ensuring adequate dynamic stability. Fig. 2 (f) visually conveys the minimum DBM values in the sagittal and lateral planes. Joint movements for both arms'  $\psi^{SP}$  and  $\psi^{ER}$  are planned into the joint space during DSP analysis, with the remaining arm joints held constant as specified for the SSP. The hip, swing leg, and arm trajectories are characterized by cubic polynomials, with the boundary conditions for these trajectories detailed in Table 1 (Sr. No. 8–12).

## 4. Many-Objective Optimization Formulation

This section presents the objective functions and constraints involved during the SSP and DSP problems of the humanoid robots.

### 4.1. Objective Functions

#### 4.1.1. Objective 1: Minimizing the average power consumption

The average power consumption ( $\mathcal{P}_{avg}$ ), as a first objective function  $f_1(\mathbf{x})$ , for the  $j^{th}$  joint over a cycle time  $\mathcal{T}_{gait}$  is computed by multiplying the motor torque ( $\tau_j(\mathbf{x})$ ) with the angular velocity ( $\dot{\phi}_j(\mathbf{x})$ ), as expressed in Eq. (5). Additionally, a value of  $K = 0.025$  is assumed to account for heat loss [94].

$$f_1(\mathbf{x}) = \sum_{j=1}^n \frac{1}{\mathcal{T}_{gait}(\mathbf{x})} \int_0^{\mathcal{T}_{gait}(\mathbf{x})} \left( \left| \tau_j(\mathbf{x}) \dot{\phi}_j(\mathbf{x}) \right| + K \tau_j(\mathbf{x})^2 \right) dt. \quad (5)$$

where  $\mathbf{x} \in X \subseteq \mathbb{R}^\kappa$  is an  $\kappa$ -dimensional decision vector with  $\kappa = 13$  for SSP and  $\kappa = 12$  for DSP. These decision variables are described in detail in Sections 4.3 and 4.4, including their respective lower and upper bounds.

#### 4.1.2. Objective 2: Maximizing the dynamic stability margin

In previous sections, the mathematical formulations have been explored for both the SSP and DSP, along with various approaches for calculating the DBM ( $X_{DBM}(\mathbf{x})$ ). Building upon the results, the aim is to optimize the DBM to enhance stability during both SSP and DSP. The second objective is expressed in Eq. (6).

$$f_2(\mathbf{x}) = \frac{1}{X_{DBM}(\mathbf{x})}. \quad (6)$$

#### 4.1.3. Objective 3: Minimizing the Fluctuation on Torque

The symbol  $\Delta\tau_{ij}^{max}(\mathbf{x})$  signifies the torque fluctuations for the  $i^{th}$  joint within the  $j^{th}$  interval and is considered as the third objective function, denoted by  $f_3(\mathbf{x})$ . A substantial value of  $f_3(\mathbf{x})$  indicates abrupt torque demands, which could potentially disrupt motion and risk motor failure. Contrary to previous studies that have treated the maximum torque fluctuation for individual joint angles as a constraint [2], this study observes that such fluctuations often occur at specific instances for certain joint sets rather than uniformly across all joints. To mitigate this, the present study computes the mean torque fluctuation across all joint angles within a cycle. By minimizing  $f_3(\mathbf{x})$ , the aim is to reduce torque variations and prevent abrupt changes in torque, which are critical for the longevity and reliability of the motors. A higher value of this objective could lead to motor overload due to sudden torque demands. The objective function ( $f_3(\mathbf{x})$ ), as formulated in Eq. (7), represents the fluctuation of torque ( $\tau^{fluc}$ ).

$$f_3(\mathbf{x}) = \text{mean}(\Delta\tau_{ij}^{max}(\mathbf{x})). \quad (7)$$

#### 4.1.4. Objective 4: Minimizing the Gait Cycle Time

Speed is a critical factor in humanoid robot locomotion. By maximizing speed, the robot ensures swift movement while adhering to all constraints. Given that the step length remains fixed at 0.06 meters,

minimizing the gait cycle time directly translates to faster locomotion. When the robot completes its gait cycle in a shorter duration, it indicates an increase in pace. The fourth objective ( $f_4(x)$ ) is expressed in Eq. (8). The gait cycle time ( $\mathcal{T}_{\text{gait}}(x)$ ) represents the duration required to complete one phase. The hip, swing leg, and arm trajectories are defined using cubic polynomial equations, ensuring smooth and synchronized motion, as detailed in Table 1. Specifically, the hip movement is governed by  $X_H(t)$ ,  $Y_H(t)$ , and  $Z_H(t)$ , while the arm movements are controlled through  $\phi_{SP}(t)$  and  $\phi_{ER}(t)$ . The initial and final times ( $t_i$  and  $t_f$ ) define the gait cycle duration, making  $\mathcal{T}_{\text{gait}}$  a direct function of these parameters.

$$f_4(x) = \mathcal{T}_{\text{gait}}(x). \quad (8)$$

## 4.2. Constraint Functions

This section outlines seven critical constraints that are integral to the robot's functional performance. Violation of these constraints could lead to undesirable robotic movements.

### 4.2.1. Constraint on Joint Violation Limit

It is imperative that the joint movements remain within their designated ranges to facilitate proper locomotion. Constraints labeled as  $g_1(x)$  and  $g_2(x)$  ensure that joint movements do not exceed these limits.

$$g_1(x) \equiv \phi^j(x) - \phi_{min}^j \geq 0, \quad (9)$$

$$g_2(x) \equiv \phi_{max}^j - \phi^j(x) \geq 0, \quad (10)$$

where  $\phi_{min}^j = -[1.14, 0.38, 1.53, 0.09, 1.19, 0.39, 0.79, 1.53, 0.10, 1.18, 0.76]^T$ , and  $\phi_{max}^j = [0.74, 0.79, 0.48, 2.11, 0.92, 0.76, 0.38, 0.48, 2.12, 0.93, 0.39]^T$ .

Here,  $\phi_{min}^j$  and  $\phi_{max}^j$  denote the minimum and maximum joint rotations, respectively, for the  $j^{th}$  joint, expressed in radians. The sequence begins with HipYawPitch and continues through the joints from the hip to the ankle for both legs, as per specifications from the official website [72].

### 4.2.2. Constraint on Minimum Balance

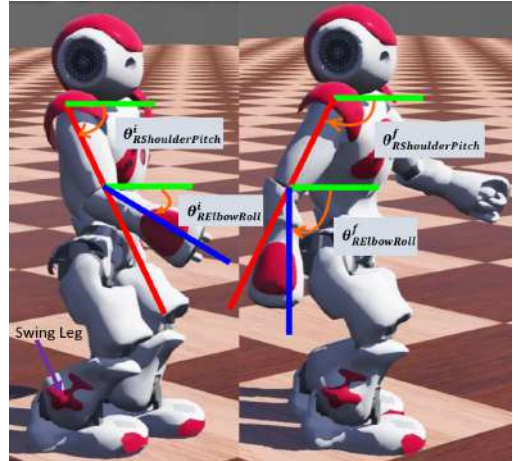
The robot's motion must exhibit dynamic balance in the  $x$  and  $y$  planes throughout its walking cycle. A positive DBM is required in both directions for each phase to prevent instability. The constraints  $g_3(x)$  and  $g_4(x)$  enforce a minimum DBM of 0.001  $m$  in both planes, as given in Eqs. (11) and (12)).

$$g_3(x) \equiv \min(X_{DBM}^i(x)) - 0.001 \text{ m} \geq 0, \quad (11)$$

$$g_4(x) \equiv \min(Y_{DBM}^i(x)) - 0.001 \text{ m} \geq 0. \quad (12)$$

### 4.2.3. Constraints on Arms Movement

The optimization of arm movements is a critical factor in enhancing the stability of humanoid robots. The  $\psi^{SP}$  and  $\psi^{ER}$  angles for both arms are considered during the optimization process. Figure 4 depicts these angles. Four decision variables are employed to control these angles precisely. The design of the arms facilitates a coordinated movement that serves to counter-balance the robot's motion. During the SSP, the right arm is programmed to swing in the opposite direction as the right leg moves forward. This action counteracts the central motion of the robot's body. Concurrently, the left arm is synchronized to move in the same direction as the swinging right leg, creating a harmonious balance in the robot's movements. To increase the dynamic stability margin, two more constraint functions have been added to ensure the correct movement in the desired direction. These constraints denoted as  $g_5(x)$  and  $g_6(x)$  in Equations (13) and (14), are designed



**Fig. 4.** Arms Movement

to synchronize the arm movements with the leg movements, thereby enhancing the robot's overall stability. Violating these constraints could result in misalignment between the arm movements and the swing leg, potentially compromising the robot's dynamic balance.

$$g_5(\mathbf{x}) \equiv \phi_{LShoulderPitch}^{final}(\mathbf{x}) - \phi_{LShoulderPitch}^{initial}(\mathbf{x}) \geq 0, \quad (13)$$

$$g_6(\mathbf{x}) \equiv \phi_{LElbowRoll}^{final}(\mathbf{x}) - \phi_{LElbowRoll}^{initial}(\mathbf{x}) \geq 0. \quad (14)$$

#### 4.2.4. Constraint on Average Dynamic Balance Margin

The average DBM in the  $y$  direction, denoted by  $g_7(\mathbf{x})$ , is crucial for maintaining a stability margin of  $0.02 m$  during SSP and  $0.05 m$  during DSP. This constraint ensures dynamic stability is preserved throughout the robot's motion.

$$g_7(\mathbf{x}) \equiv \min Y_{DBM}^{Avg} m - Y_{DBM}^{Avg}(\mathbf{x}) \geq 0. \quad (15)$$

### 4.3. Single Support Phase (SSP) Scenario

The MaOP for the SSP is presented below. The objective functions and constraints have been defined in the previous section.

$$\text{minimize}_{\mathbf{x} \in X \subseteq \mathbb{R}^{13}} \mathcal{P}_{\text{avg}} \text{ ( refer to Eq. (5) ),}$$

$$\text{minimize}_{\mathbf{x} \in X \subseteq \mathbb{R}^{13}} \frac{1}{X_{DBM}} \text{ ( refer to Eq. (6) ),}$$

$$\text{minimize}_{\mathbf{x} \in X \subseteq \mathbb{R}^{13}} \tau^{\text{fluc}} \text{ ( refer to Eq. (7) ),}$$

$$\text{minimize}_{\mathbf{x} \in X \subseteq \mathbb{R}^{13}} \mathcal{T}_{\text{gait}} \text{ ( refer to Eq. (8) ),}$$

subject to Constraints  $g_1(\mathbf{x})$  through  $g_6(\mathbf{x})$  given in Eqs. (9) through (14),

$$g_7(\mathbf{x}) \equiv 0.02 m - Y_{DBM}^{Avg}(\mathbf{x}) \geq 0, \quad (16)$$

$$x_i^L \leq x_i \leq x_i^U,$$

with  $x_i^L = [ 0.20, -0.05, -0.05, 0.001, 0.001, -0.2, -0.2, 0.015, 0.4, 0, 1, 0.04, 0.2 ]^T$ , and  $x_i^U = [ 0.31, 0.05, 0.05, 0.2, 0.2, 0.2, 0.2, 0.030, 4, \pi/2, 2, 0.5, 1.5 ]^T$ ,

There are a total of 13 decision parameters in this scenario.  $x_1$  represents the hip height, which directly influences the robot's stability and stride length.  $x_2$  and  $x_3$  correspond to the start and end velocities associated with the hip height. These parameters control how smoothly the robot transitions between different phases of the gait cycle.  $x_4$  and  $x_5$  denote the start and end sagittal velocities, affecting forward and backward movements. These velocities impact the robot's overall speed and efficiency.  $x_6$  and  $x_7$  represent the start and end lateral velocities, influencing side-to-side movements. Proper lateral motion ensures balance during walking and turning.  $x_8$  and  $x_9$  are critical for the robot's swing phase.  $S_h^{max}$ , the maximum swing height, determines how high the robot's foot lifts during each step. The duration spent in the SSP is controlled by  $x_9$ . Additionally, the left arm's motion is captured by  $x_{10}$  and  $x_{11}$ , representing the initial and final  $\psi^{SP}$  angles. These angles dictate the arm's position during the gait cycle. Lastly,  $x_{12}$  and  $x_{13}$  denote the initial and final  $\psi^{ER}$  angles for the left arm. These angles influence the arm's orientation and movement.

#### 4.4. Double Support Phase (DSP) Scenario

The MaOP is presented for the double support phase, as given below.

$$\begin{aligned}
& \underset{x \in X \subseteq \mathbb{R}^{12}}{\text{minimize}} \quad \left\{ \mathcal{P}_{\text{avg}}, 1/X_{\text{DBM}}, \tau^{\text{fluc}}, \mathcal{T}_{\text{gait}} \right\} \quad (\text{refer to Eqs. (5) to (8)}), \\
& \text{subject to the constraints } g_1(x) \text{ through } g_6(x) \text{ given in Eqs. (9) through (14),} \\
& \quad g_8(x) \equiv 0.05 m - Y_{\text{DBM}}^{\text{Avg}}(x) \geq 0, \\
& \quad x_i^L \leq x_i \leq x_i^U,
\end{aligned} \tag{17}$$

with  $x_i^L = [0.20, -0.05, -0.05, 0.001, 0.001, -0.2, -0.2, 0.2, 0, 1, -0.50, -1.50]^T$ , and  $x_i^U = [0.31, 0.05, 0.05, 0.2, 0.2, 0.2, 0.2, 2, \pi/2, 2, -0.04, -0.2]^T$ .

In this DSP scenario, there are 12 decision parameters. Except for the maximum swing height, the first seven and last five decision parameters remain consistent with those in the SSP scenario.

### 5. The Proposed Optimization Approach

The EMaO methodology, specifically the NSGA-III algorithm, has been employed to solve the SSP and DSP problems of the NAO humanoid robot for four conflicting objectives: minimizing  $\mathcal{P}_{\text{avg}}$ , maximizing  $X_{\text{DBM}}$ , minimizing  $\tau^{\text{fluc}}$ , and minimizing  $\mathcal{T}_{\text{gait}}$ . Solving MaOPs using EMaO algorithms necessitates achieving ND solutions that exhibit two crucial goals: 1) obtain ND solutions with high convergence and 2) preserve a diverse set of solutions with a uniform distribution that covers different regions of the solution space. However, maintaining diversity alongside securing a converged Pareto-optimal (PO) front becomes increasingly difficult in high-dimensional problems. The NSGA-III algorithm [27] outperforms its predecessor, NSGA-II, in MaOPs by effectively preserving diversity and ensuring convergence [20]. This enhancement stems from using uniformly distributed reference points. NSGA-III enhances the offspring creation process by selecting ND solutions near reference vectors in each generation. This facilitates a more diverse solution set. NSGA-III's adaptability to constrained MaOPs was extended in [28]. Five independent runs of the NSGA-III algorithm are conducted to ensure high confidence in the optimality of the solutions obtained. For both SSP and DSP, distinct MaOPs have been formulated to optimize all four objectives simultaneously. However, before optimizing these objectives simultaneously, ensuring that they contradict each other is essential. Therefore, initially, a single-objective optimization (SOO) of these four objectives, taken one objective at a time, was performed using a real parameter Genetic Algorithm (GA). The trade-off among these objectives is visible, as presented in Table 2.

Optimizing a single objective at a time for a robot's gait could lead to trade-offs that may not be desirable. For instance, a robot walking at a slower speed with the higher hip height and lower swing height would consume less power. However, a lower hip height with a negative slope and higher swing height could improve stability. When power consumption was reduced, arm movements were not significant, but they moved significantly to increase stability. To maintain the higher stability, the torque fluctuation was very high, leading to increased power consumption. Although power consumption changes with torque fluctuations, when torque fluctuations were minimized, the power consumption was slightly higher, with the moderate stability. When power was minimized, the torque fluctuations were moderate, but stability was poor as the robot walked very slowly. Similarly, when the gait cycle time was minimized, the torque fluctuations were moderate, but power consumption was higher, as the robot was walking at a high speed. These observations indicate that the objectives of power consumption, DBM (stability), torque fluctuation, and gait cycle time were contradictory to each other. Optimizing one objective at a time might lead to undesirable trade-offs in the other objectives. Therefore, it is essential to consider these four objectives simultaneously in a MaOO framework rather than optimizing them as separate single-objective functions. This approach would ensure a more balanced and efficient solution.

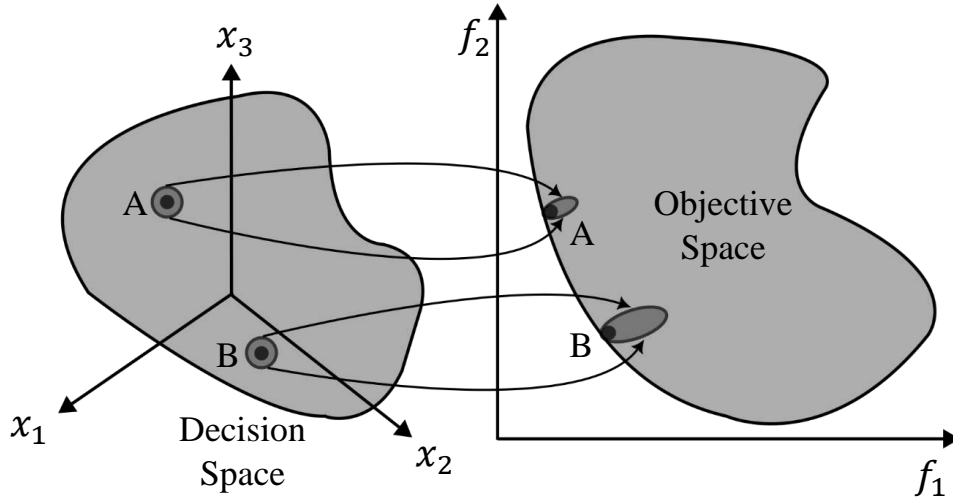
Table 2: Single-objective optimization results using a GA.

Decision Variables	Gait Parameters				Unit
$h$	0.31	0.24	0.21	0.20	m
$V_s^Z$	-0.002	-0.042	0.499	0.004	m/sec
$V_e^Z$	0.002	-0.049	0.029	0.023	m/sec
$V_s^X$	0.002	0.043	0.020	0.199	m/sec
$V_e^X$	0.007	0.109	0.055	0.001	m/sec
$V_s^Y$	0.022	0.087	0.087	0.193	m/sec
$V_e^Y$	-0.027	-0.200	-0.089	-0.19	m/sec
$S_h^{max}$	0.015	0.018	0.015	0.019	m
$q_i^{ShoulderPitch}$	1.56	0.587	1.027	1.18	rad
$q_f^{ShoulderPitch}$	1.58	1.21	1.038	1.57	rad
$q_i^{ElbowRoll}$	0.30	0.45	0.211	0.28	rad
$q_f^{ElbowRoll}$	0.31	0.47	0.221	0.57	rad
$\mathcal{P}_{avg}$	1.14	6.47	5.09	9.38	$W$
$1/X_{DBM}$	16.51	12.16	14.90	16.36	$m^{-1}$
$\tau^{fluc}$	0.069	0.23	0.064	0.11	N-m
$\mathcal{T}_{gait}$	4	2.20	1.27	0.83	sec

## 6. Methodology for Robustness Analysis

In contrast to single objective optimization problems (SOPs) that aim for a singular optimal solution, EMO methods strive to identify a collection of PO solutions for multi-objective optimization problems (MOPs) [95]. These PO solutions represent the globally superior options within the viable decision space. On the PF, enhancing one objective necessitates the compromise of at least one other objective. A key challenge with many EMO approaches is that the PO solutions might be highly susceptible to minor changes in the decision variable vectors. Even small changes in the values of the decision variables could significantly alter the performance of one or more objectives for a PO solution, making the solutions less robust and reliable in practical applications. The primary aim of EMO algorithms has been to acquire a set of PO solutions that are both diverse and well-distributed [95, 96]. While these ND solutions are considered the best from a theoretical standpoint in optimization, they might be highly susceptible to minor changes in the vicinity of the decision variables for one or more PO solutions. In practice, the PO solutions could not be directly adopted due to the unpredictability associated with executing and applying these solutions in their intended forms. A slight deviation in implementing a desired solution could result in an objective vector that diverged from the original goal. These small perturbations in the solution might make it suboptimal, infeasible, improved, or remain ND, when solving multi- and MaOPs. Thus, practitioners typically seek a collection of robust solutions that demonstrate reduced sensitivity to small changes in the neighborhood of the solution space.

Figure 5 graphically depicts the sensitivity of two decision vectors, A and B, within the objective space. It is evident that a slight perturbation at decision vector B leads to a significant alteration in the objective values, whereas a similar perturbation at decision vector A results in a minimal change. Therefore, solution A is said to be more robust than solution B. In multi-objective optimization (MOO), robustness is defined differently from SOO due to the presence of two or more objectives and a set of ND solutions. This understanding of robustness is essential for developing optimization strategies that are resilient to variations and uncertainties in the decision-making process. Robustness is a concept that has been interpreted in various ways by different researchers. It is a principle that had been extensively explored and applied within the realms of both SOPs and MOPs. Branke [97] identified that robust solutions necessitate a substantial increase in computational complexity due to the need for evaluating a larger number of solutions. Subsequently, he [98] proposed modifications



**Fig. 5.** Comparative Sensitivity Analysis of Decision Vectors A and B in Multi-Objective Optimization Space.

to the EMO formulation that efficiently integrated robustness without a substantial increase in computational complexity. The SOP was reformulated as an MOP with the aim of minimizing both the objective function and the robustness measure to identify the robust optimal solution [99]. Tsutsui and Ghosh [100] introduced a robust solution-searching scheme within genetic algorithms to assess robust solutions in SOPs.

In [101], the ranking method used to determine the selection probabilities for identifying robust solutions had been redefined for both SOPs and MOPs. Deb and Gupta [102] introduced a pair of methodologies for determining the robust frontier across four test scenarios, utilizing the NSGA-II algorithm [103]. The first method focused on minimizing the average effective function rather than the original function, while the second method introduced a robustness constraint into the restructured MOP. Furthermore, these methods had also been adapted to accommodate constraints [104]. Zhou et al. [105] introduced an approach to robustness in the context of the MOP that utilized a Gaussian process model, aiming to lessen the computational load in the presence of interval uncertainty. The efficacy of proposed algorithms was evaluated by examining their convergence, the extent of their coverage, and the frequency of successfully achieving robust solutions, as produced by robust multi-objective algorithms [106]. Mirjalili and Lewis [107,108] examined the prevalent issues in test problems and introduced a methodology for creating robust multi-objective test problems, which allowed for the adjustment of various characteristics and levels of complexity. Yadav et al. [20] incorporated uncertainty handling through the use of NSGA-III. However, the test problems they employed did not include any functional constraints.

The robustness for MaOP using NSGA-III was implemented in [20] for different test problems. The NSGA-III algorithm, with its enhanced selection scheme, association operator, and niche preservation mechanism, significantly advanced the convergence and coverage of MaOPs. These improvements render NSGA-III particularly adept at tackling MaORO problems, ensuring robust solutions that are well-adapted to a variety of uncertain conditions. This makes NSGA-III the preferred choice for obtaining reliable and high-quality solutions in complex optimization scenarios. Yadav et al. [20] conducted a comparative analysis of PO solutions for a five-objective MaOP, such as the river pollution problem, using NSGA-II and NSGA-III. Their findings revealed that NSGA-III yielded the superior performance, with ND solutions demonstrating enhanced distribution. While the literature features numerous robust optimization studies for MOPs, research on robustness within MaOO, particularly involving more than three objectives to address the gait cycle challenges of humanoid robots, remains scarce. The innovative aspect of this research lies in its application of NSGA-III to manage uncertainty in MaOO, incorporating multiple constraints. This approach effectively handles parameter uncertainties while solving constrained MaOPs. In the context of MOPs, robust optimization was first explored in [102]. The MOP is formulated, as shown in Eq. (18):

$$\begin{aligned} & \min_{\mathbf{x}} \{f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_M(\mathbf{x})\}, \\ & \text{subject to } \mathbf{x} \in \mathcal{X} \end{aligned} \quad (18)$$

where  $\mathcal{X}$  denotes the feasible search space. A solution  $\mathbf{x}$  is termed robust if the corresponding objective vector  $f(\mathbf{x})$  remains relatively unaffected, or is only minimally sensitive, to small perturbations within the vicinity of  $\mathbf{x}$ . In the realm of robust SOO, two prevalent methodologies are cited in the literature [98]. These methodologies had been adapted to establish a robust PF in MOO scenarios [102]. The adaptations of Robust Solutions of Type I and Type II for MOO as defined in [102] are presented as follows:

**Robust Solution of Type I:** This approach optimizes the mean effective objective function ( $f^{\text{eff}}$ ) instead of the original one. Within an  $\alpha$ -neighborhood ( $\mathcal{U}_\alpha(\mathbf{x})$ ), a solution  $\mathbf{x}^*$  is deemed a Type I multi-objective robust solution, if it stands as a globally feasible PO solution to the multi-objective minimization problem defined as:

$$\begin{aligned} & \min_{\mathbf{x}} \left\{ f_1^{\text{eff}}(\mathbf{x}), f_2^{\text{eff}}(\mathbf{x}), \dots, f_M^{\text{eff}}(\mathbf{x}) \right\}, \\ & \text{subject to } \mathbf{x} \in \mathcal{X}, \end{aligned} \quad (19)$$

where  $f_j^{\text{eff}}$  is defined as follows:

$$f_j^{\text{eff}} = \frac{1}{\mathcal{U}_\alpha(\mathbf{x})} \int_{y \in \mathcal{U}_\alpha(\mathbf{x})} f_j(y) dy. \quad (20)$$

**Type II Robust Solution:** Here, the focus is on the normalized difference between the perturbed and the original objective functions, using it as a constraint for more precise robust solution control. A solution  $\mathbf{x}^*$  is recognized as a Type II robust solution in MOO, when it is a globally feasible PO solution according to the following multi-objective minimization criteria:

$$\begin{aligned} & \min_{\mathbf{x}} \{f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_M(\mathbf{x})\}, \\ & \text{subject to} \\ & \frac{\|\mathbf{f}^{\text{eff}}(\mathbf{x}) - \mathbf{f}(\mathbf{x})\|_2}{\|\mathbf{f}(\mathbf{x})\|_2} \leq \epsilon, \\ & \mathbf{x} \in \mathcal{X}. \end{aligned} \quad (21)$$

The effective objective function corresponding to  $\mathbf{x}$  is determined by averaging  $N$  original objective functions, each assessed from  $N$  distinct solutions within the  $\alpha$ -vicinity of  $\mathbf{x}$  given in (20). On the other hand, the formulation described by (21) optimizes the original multi-objective function within the feasible decision space  $\mathcal{X}$  while incorporating an additional constraint. This constraint considers only those solutions that satisfy a normalized change in the objective vector below a user-defined threshold  $\epsilon$ . Alongside the constraint handling strategy, alternative forms of the constraint specified in (21) had been proposed and implemented, as documented in [104]. The  $f^{\text{eff}}$  is computed using  $N$  neighboring solutions within the  $\alpha$ -neighborhood of each reference point in NSGA-III. This  $f^{\text{eff}}$  can be incorporated into the robustness definitions for Type I and Type II, as defined by (20) and (21), respectively. In the subsequent section, NSGA-III has been implemented to solve the 25-DOF NAO robot's gait generation problem.

## 7. Results and Discussion

The selection of NSGA-III for analyzing SSP and DSP scenarios is preceded by a comprehensive comparative analysis against three state-of-the-art EMaO algorithms: the multi-objective evolutionary

algorithm based on decomposition (MOEA/D) [109], the strength Pareto evolutionary algorithm 2 (SPEA2) [110], and the hypervolume-based evolutionary algorithm (HypE) [111]. These algorithms are evaluated using the PlatEMO platform [112], where they are referred to as DCNSGA-III (Dynamic Constrained NSGA-III), C-MOEA/D (Constrained MOEA/D), HypE, and SPEA2. Each algorithm is applied separately to address the gait generation problem during SSP and DSP. The comparison is performed during SSP for the variable  $x_i$ . Perturbations are considered at 10% or greater than the lower and upper bounds of the decision variables' ranges. A perturbation of 10% demonstrates that even with uncertainties approaching this threshold, including modeling errors, the optimal solutions exhibit minimal variance. The applied perturbations for the  $i$ th decision variables are as follows:  $\alpha_i = 0.02$  for  $i = 1, 4, 5$ ;  $\alpha_i = 0.01$  for  $i = 2, 3$ ;  $\alpha_i = 0.04$  for  $i = 6, 7$ ;  $\alpha_i = 0.002$  for  $i = 8$ ;  $\alpha_i = 0.2$  for  $i = 9, 10, 11$ ;  $\alpha_i = 0.1$  for  $i = 12$ ; and  $\alpha_i = 0.15$  for  $i = 13$ . The  $\alpha$  value depends on the real-time tolerance and how much control is available on one decision variable. To compute the  $f^{\text{eff}}$ ,  $N = 16$  solutions are generated in the vicinity of a solution. The neighboring solutions are created using the Taguchi array method, which employs 2 levels and 13 factors. Similarly, comparisons are conducted for the DSP. The perturbation parameters  $\alpha_i$  for the initial seven and final five variables are maintained identically to those in the SSP. This happens because the eighth variable, representing the swing height, is exclusive to the SSP and thus, is not applicable within the DSP context. Type II robustness, on the other hand, considers an inequality constraint on the normalized difference between the original solution and the  $f^{\text{eff}}$  solution by introducing the parameter  $\epsilon$ . In this study, a value of  $\epsilon = 0.25$  is used.

The comparative assessment is conducted for both robustness types (Type I and Type II) in SSP and DSP scenarios, with each algorithm evaluated using two performance indicators: hypervolume (HV) [113] and constraint violation percentage, as shown in Table 3. All objective values are normalized between 0 and 1, and the reference point for HV computation is set to  $[1, 1, 1, 1]$ , with HV calculations performed using the Pymoo library [114]. To ensure reliability, each configuration is executed twice, and the average values are used for analysis.

In the Type I robustness evaluation, constraint violation computations exclude the arm movement constraints ( $g_5(x)$  and  $g_6(x)$ ), since all algorithms struggle to satisfy them. While these constraints are essential for synchronizing arm and leg movements to achieve a human-like motion pattern, their violation does not significantly impact fundamental locomotion. Consequently, constraint violation percentages are computed based on the decision variable vectors obtained in the final generation, omitting these specific constraints. However, in the Type II robustness evaluation, all constraints, including  $g_5(x)$  and  $g_6(x)$ , are considered in the computation. This decision is based on the observation that Type II exhibits better constraint-handling capabilities, a point further elaborated in the subsequent discussion.

In the SSP Type I scenario, NSGA-III demonstrates strong performance, achieving an average HV of 0.583, while maintaining a moderate constraint violation of 21.25%. Its performance improves in SSP Type II, where all constraints are considered, resulting in an increased average HV of 0.607, after satisfying all constraints (0% violation). The algorithm exhibits even better results in DSP scenarios, attaining high HV values of 0.707 in Type I and 0.778 in Type II. Notably, in DSP Type II, NSGA-III demonstrates exceptional performance by satisfying all constraints while securing the second-highest HV among all evaluated algorithms. In contrast, MOEA/D demonstrates excellent constraint-handling capabilities, achieving 0% violations in multiple scenarios. However, it consistently yields lower HV values than NSGA-III.

SPEA2 achieves notably high average HV values (reaching 0.825 in DSP Type II), but its solutions are predominantly infeasible due to severe constraint handling issues. In SSP scenarios, SPEA2 exhibits 100% constraint violations for both Type I and II, indicating that none of its solutions satisfy the problem constraints despite the high HV values. While its constraint handling improves slightly in DSP Type I (around 34% violations), it deteriorates significantly in DSP Type II with violations exceeding 90%. Similarly, HypE demonstrates substantial constraint handling deficiencies, with 100% violations in SSP scenarios and over 90% violations in DSP Type II. Its HV values are generally lower than those of other algorithms and show high variability between the runs (e.g., ranging from 0.346 to 0.654 in DSP Type II), making it the least reliable choice among all evaluated algorithms.

NSGA-III’s consistent performance across all test scenarios, along with its ability to balance HV maximization and constraint satisfaction, justifies its selection for a detailed SSP and DSP analysis. The algorithm demonstrates particular strength in Type II scenarios, where it achieves perfect constraint satisfaction after maintaining competitive HV values. This balanced performance across both evaluation criteria establishes NSGA-III as a reliable choice for addressing the complexity and constraints inherent in SSP and DSP optimization problems.

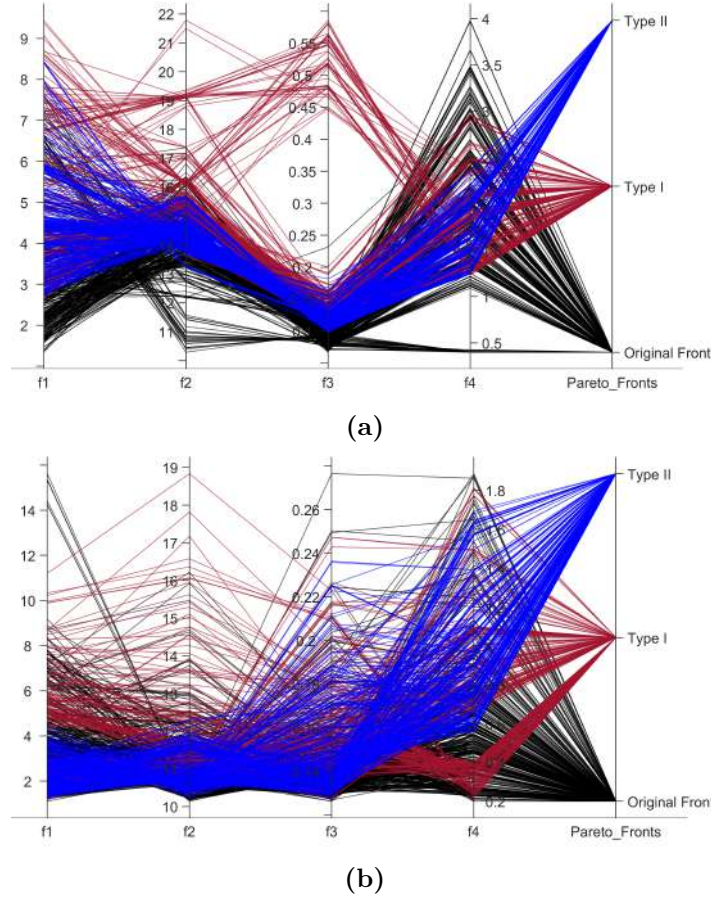
Table 3: Comparative Performance of Algorithms Across Robustness Types and Scenarios

Scenario	Robustness Type	Algorithm	Run	Hypervolume	Constraints Violation (in %)
SSP	Type I	NSGA-III	Run 1	0.490	19.17
			Run 2	0.675	23.33
		MOEA/D	Run 1	0.137	7.50
			Run 2	0.163	16.67
		SPEA2	Run 1	0.668	100
	Run 2		0.617	100	
	HypE	Run 1	0.349	100	
		Run 2	0.543	100	
	Type II	NSGA-III	Run 1	0.596	0.00
			Run 2	0.617	0.00
MOEA/D		Run 1	0.571	0.00	
		Run 2	0.528	0.00	
SPEA2		Run 1	0.737	100	
	Run 2	0.655	100		
HypE	Run 1	0.455	100		
	Run 2	0.407	100		
DSP	Type I	NSGA-III	Run 1	0.700	10.84
			Run 2	0.713	15.03
		MOEA/D	Run 1	0.522	0.00
			Run 2	0.547	0.00
		SPEA2	Run 1	0.727	36.33
	Run 2		0.760	32.33	
	HypE	Run 1	0.397	41.33	
		Run 2	0.448	31.67	
	Type II	NSGA-III	Run 1	0.793	0.00
			Run 2	0.763	0.00
MOEA/D		Run 1	0.586	0.00	
		Run 2	0.589	0.00	
SPEA2		Run 1	0.819	91.67	
	Run 2	0.831	94.00		
HypE	Run 1	0.346	99.33		
	Run 2	0.654	94.00		

Therefore, the NSGA-III algorithm is employed to investigate the MaOO of the gait cycle for a 25-DOF NAO robot. The results obtained from NSGA-III are presented separately for SSP and DSP

to highlight the distinct characteristics and performance metrics of each phase. Type I and Type II robust fronts are determined using parameters  $(\alpha, N, \epsilon)$  within the NSGA-III framework. Initially, the algorithm is tested with various population sizes and generations using a trial-and-error approach, revealing that a population of 200 and 200 generations yields satisfactory results. This selection aligns with the settings used in prior studies [115,116]. Subsequently, NSGA-III is executed five times with 200 generations and a population size of 200. Constraints in NSGA-III are handled as proposed by Jiao [29], which proves to be more effective in managing constraints. Reference directions are generated using an incremental lattice design strategy, as proposed by Takagi [117].

Simulations are executed during SSP and DSP using the platEMO platform. The PF obtained from the simulations corresponds to different types of robustness, namely Type I and Type II, along with the original PF ( $\mathcal{PF}^{\text{org}}$ ). Figure 6 presents a parallel coordinate plot (PCP) depicting these fronts.

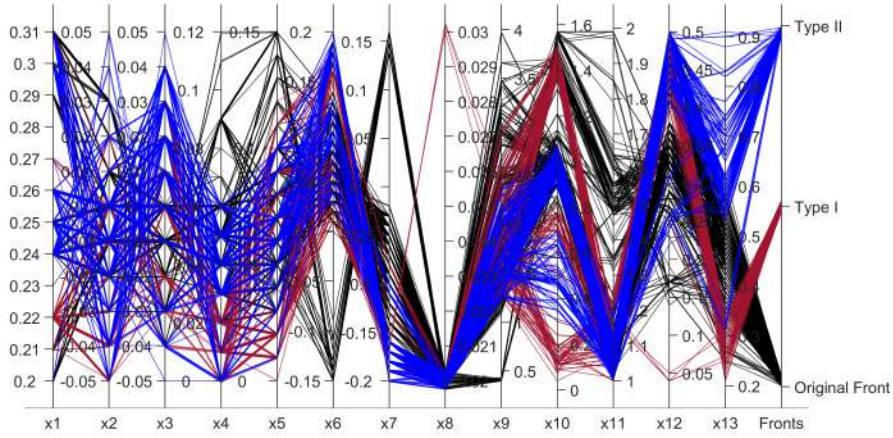


**Fig. 6.** Parallel coordinate plot showing objective space for original fronts along with Type I and II robust fronts during (a) Single Support Phase. (b) Double Support Phase.

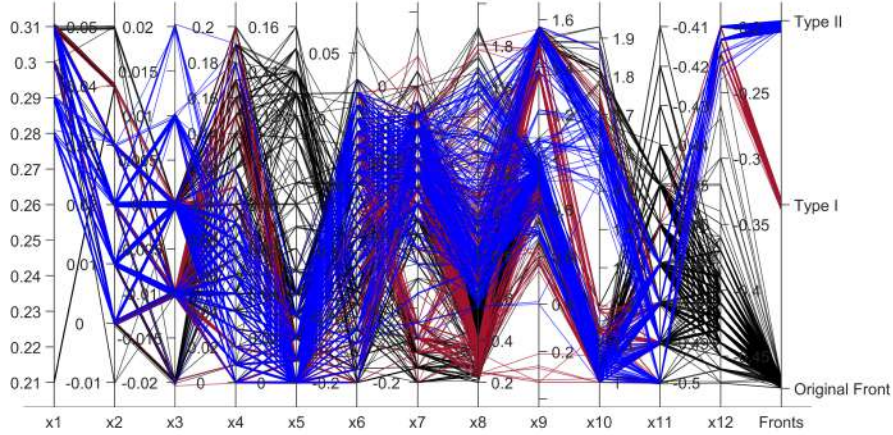
The  $\mathcal{PF}^{\text{org}}$  represents the ND solutions obtained without any perturbation, combined from multiple runs. For Type I robustness, the optimization considers the  $f^{\text{eff}}$ . The obtained ND solutions are used to find the actual objective values, which are then plotted in the figure. During the implementation of Type I robustness, after introducing perturbations, all  $N$  solutions are checked for constraint violations before optimizing the  $f^{\text{eff}}$ . However, after finding the actual objective values, it is discovered that some solutions did not satisfy all the constraints, with approximately 90% of the ND solutions failing to satisfy the arms movements in the correct order constraint. This issue can arise because the  $f^{\text{eff}}$  are optimized, and the direction suggested by the NSGA-III algorithm is optimal for the  $f^{\text{eff}}$  but not necessarily for the actual objective functions, leading to constraint violations. Meanwhile, Type II robustness is found to satisfy all the constraints, including those that are not satisfied during the implementation of Type I robustness. The combined results are plotted in Figures 6a for the SSP and 6b for the DSP objective spaces, as the PCPs. In these PCPs, the four objectives ( $f_1$  to  $f_4$ ) are plotted

on four parallel axes, with an additional axis showing the different types of fronts: Original, Type I, and Type II. The first objective,  $f_1$ , represents the  $\mathcal{P}_{avg}$ ; the second objective,  $f_2$ , is the  $X_{DBM}$ ; the third objective,  $f_3$ , is the  $\tau^{fluc}$ ; and the fourth objective,  $f_4$ , is the  $\mathcal{T}_{gait}$ .

During the SSP, in the case of Type I and Type II robustness, the robust fronts got shifted away from the original ND front and moved inside the objective space. This indicates a trade-off between robustness and optimality. The robust fronts prioritized solutions that are less sensitive to perturbations, whereas the original front represents optimal solutions. Objective  $f_2$  exhibited two regions of stability: one with the higher stability at high power consumption and a faster pace, and another region corresponding to moderate to poor stability. However, neither Type I nor Type II robustness could find solutions near the first region, as a slight perturbation could result in an inefficient solution or a solution that violates the constraint functions. The maximum and minimum ranges of all the objectives have been reduced in the Type I and Type II fronts, except that Type I is able to find a different region of faster locomotion with moderate power but at higher torque fluctuation. The  $f_4$  objective also has two regions: one corresponds to the faster locomotion (consuming around 0.4 seconds), and another consists of moderate speed to slow speed (from 1 to 4 seconds).



(a)



(b)

**Fig. 7.** Parallel coordinate plot showing decision space with original, Type I and Type II robustness solutions during (a) Single Support Phase. (b) Double Support Phase.

Again, neither Type I nor Type II robustness could find solutions near the first region of faster locomotion. When the gait cycle time is minimized as an SOO, and other objectives had their worst values (refer to Table 2), Type II robustness found solutions that are slightly slower (taking 0.4 seconds extra) but kept the other objectives in line, achieving comparatively good stability with low power consumption and torque fluctuations. The range of the  $f_4$  objective has been considerably reduced during Type II robustness compared to other fronts. During DSP, a portion of the robust

front coincides with the  $\mathcal{PF}^{\text{org}}$ , particularly for objectives  $f_1$  to  $f_3$ . However, the range of the  $f_4$  objective has been considerably reduced in the Type II robust front. This overlapping is attributed to the fact that during the DSP, the robot has a larger support polygon, enabling it to produce different trajectories that can satisfy constraints and identify solutions that are more robust while maintaining reasonable objective values. The Type II robustness approach seems to provide a better trade-off between robustness and optimality, as it not only considers the  $f^{\text{eff}}$  but also incorporates an additional constraint to ensure that the solutions remain within a specified tolerance of the original objective functions. This approach helps find solutions that are both robust to perturbations and sacrifice some optimality in the objective values.

Figure 7 shows the PCP between the gait parameter solutions responsible for producing the robust front in the objective space. Figures 7a and 7b depict the PCPs for the SSP and DSP, respectively. It can be observed that there are more different hip height options available during the SSP compared to the DSP because a higher hip height with a straight trajectory is found to help in reducing power consumption. However, in the case of the SSP, the support polygon is much smaller than in the DSP, and to satisfy all the constraints, a variety of different hip heights are required. Not only are there more different hip heights, but there are also more diverse hip trajectory types, such as positive slope, negative slope, and straight trajectories, available during the SSP than in the DSP. The DSP consists of fewer nodes for hip trajectories, mostly comprising positive slope trajectories, as they help in reducing power consumption. This happens due to the shorter distance traveled during the DSP compared to the SSP.

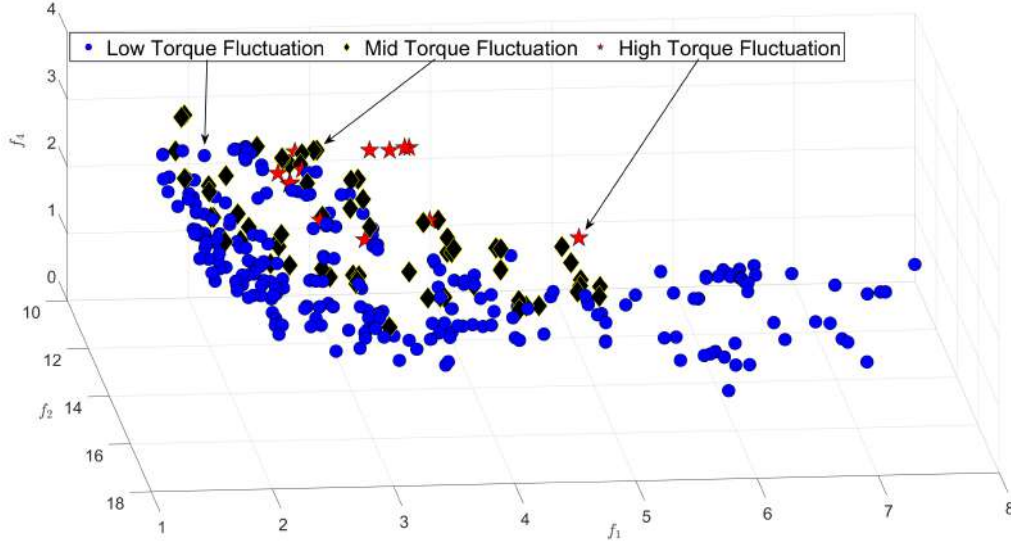
The initial and final sagittal velocities in each phase are very limited in the case of robust solutions compared to the original front. The robust end sagittal velocities during the DSP are significantly lower than the original ND solutions because this velocity would be transferred to the SSP, and a low velocity during transition helps in reducing the impact and fluctuation of torque. The end lateral velocity during the SSP is much higher compared to DSP for Type II robust solutions because this would be the initial lateral velocity during the DSP, and as both legs are on the ground, it can take a higher velocity while satisfying all the constraints. Type I, Type II, and the original fronts have mostly found ND solutions with very low swing heights. However, Type II robust solutions contained a few solutions offering the higher swing heights. This can be used to negotiate obstacle. The only issue is that the arm movements are not in the correct order, but this can be neglected because it has a very less effect on stability. The  $\psi^{\mathcal{SP}}$  and  $\psi^{\mathcal{ER}}$  arm joints have much higher values during the SSP compared to the DSP, because the arms are mostly moving in the sagittal plane, and the robot is primarily moving in the lateral plane during the DSP. In the next section, a deeper understanding of how the gait parameters can affect the objective values will be gained.

## 7.1. Single Support Phase

Figure 8 shows the scatter plot of the objective space without perturbation for the SSP. A global Pareto set is considered by running the NSGA-III algorithm multiple times. All ND sets from multiple runs are combined, and the best ND solutions are regarded as the  $\mathcal{PF}^{\text{org}}$ . It is noteworthy that the robust front obtained upon solving the SSP example using NSGA-III is well-distributed.

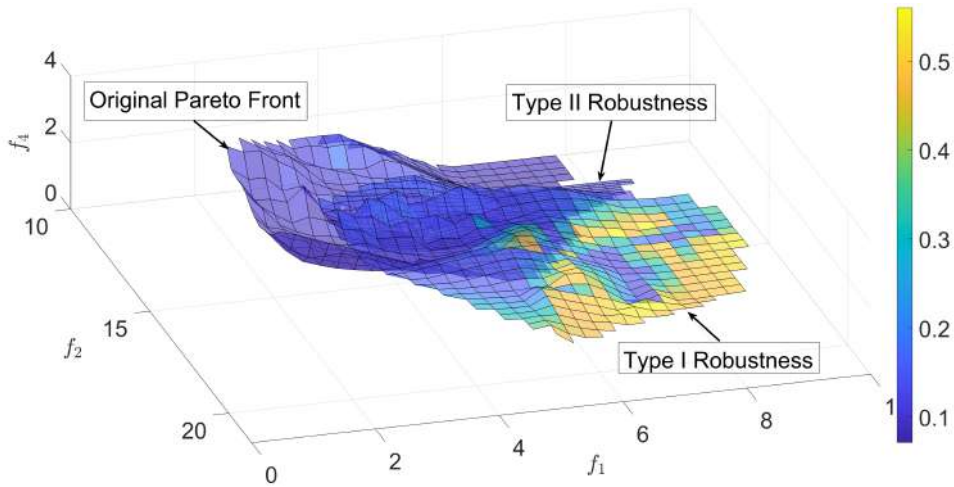
In the scatter plot,  $f_1$ ,  $f_2$ , and  $f_4$  are shown on the  $x$ ,  $y$ , and  $z$ -axes, respectively, while  $f_3$  is divided into three groups: low, medium, and high torque fluctuations. The circle marker represents low torque fluctuation, the diamond marker represents medium torque fluctuation, and the star marker represents high torque fluctuation points. There are numerous points with low torque fluctuations throughout the PF. However, medium and high torque fluctuations mostly occurred in the high stability region, resulting from the robot’s quick movements. Power consumption is lower, when the gait cycle time is higher, indicating a slower speed, while at faster paces, power consumption increases. Notably, at faster speeds, there are many points with high stability and low torque fluctuations. Additionally, there are a few points with high stability and high torque fluctuations but low power consumption. These solutions are dependent on specific requirements. However, the most crucial aspect is their implementation, as a slight perturbation might cause a variance from the intended results. Type I and Type II robustness approaches helped in finding fronts that sacrificed a little optimality but are more robust to perturbations. By considering the mean effective objective functions (Type I)

or introducing an inequality constraint on the normalized difference between the original and mean effective objective functions (Type II), these approaches aimed to identify solutions that are less sensitive to minor changes in the decision variables, while maintaining reasonable objective values.



**Fig. 8.** Scatter plot of original Pareto-front during single support phase

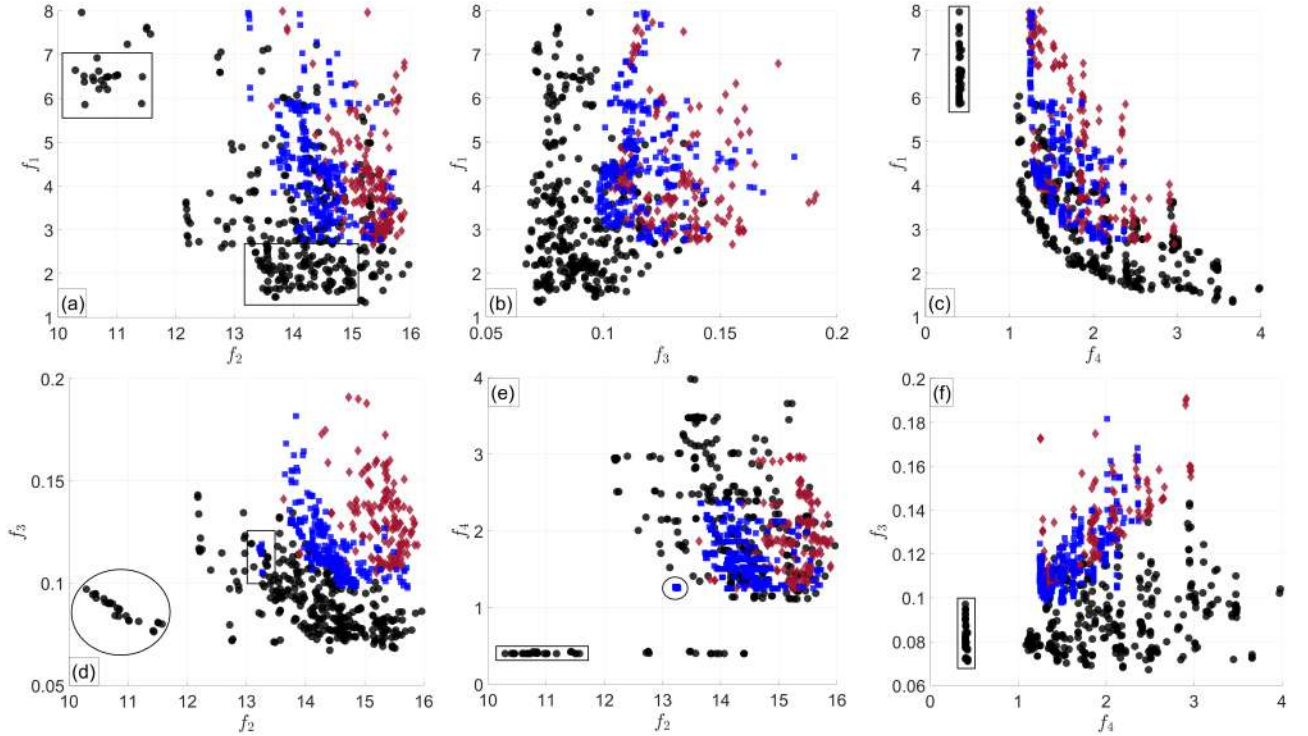
The surface plot for the SSP in Figure 9 illustrates complex trade-offs between power consumption, stability, gait cycle time, and torque fluctuation in humanoid robot gait optimization. The  $\mathcal{PF}^{\text{org}}$  spans a wide range in the objective space, offering diverse solutions with clear trade-offs between objectives. Type I robustness shifts into the objective space, covering regions different from the original front. It provides robust solutions, but at the cost of higher torque fluctuations, as indicated by the warmer colors on the plot. Some of these solutions violate arm movement constraints. In contrast, Type II robustness occupies a more compact region in the objective space. It generates solutions with lower torque fluctuations, as shown by the cooler colors, and provides better robust solutions without violating constraints. Type II solutions are more concentrated in the lower power consumption and moderate stability region. This analysis suggests that the Type II approach is superior, offering robust solutions that maintain feasibility and achieve better optimal solutions making them more suitable for real-world application in humanoid robot gait optimization.



**Fig. 9.** Surface plot comparing the original Pareto-front with Type I and Type II robust solutions for the single support phase.

Figure 10 shows the scatter plots of two objectives at a time for the SSP. The circle markers

represent the  $\mathcal{PF}^{\text{org}}$ , the diamond markers represent Type I robustness, and the square markers represent Type II robustness. In Figure 10(a), the scatter plot between objectives  $f_1$  and  $f_2$  exhibits two knee regions (KRs), enclosed by rectangles. KRs signify that leaving these regions would result in the higher degradation in one objective for a slight benefit in the other, making them highly valuable for DMs. The first KR has very few points, indicating the higher stability at the expense of high power consumption. Notably, neither Type I nor Type II robust points are available near this first KR. Both robust fronts are found near the second KR and move inside the objective space. The Type II robust front is better than the Type I robust front in terms of better stability and minimum power consumption. This discrepancy can arise because, at the higher speeds, there are fewer trajectories available that can enhance stability. Additionally, it has been found that there are very few solutions between these two KR, mainly because it is not possible to find suitable gait parameters without violating the constraints imposed in the optimization process. In Figure 10(b), objectives  $f_1$  and  $f_3$  seem to be minimal for similar gait parameters. However, the robust fronts do not lie very close to the  $\mathcal{PF}^{\text{org}}$  but maintain an offset and move towards the top-right in the objective space, indicating that the robust front is not close to the optimal PF. Both the Type I and Type II robust fronts show a good trade-off between power consumption and torque fluctuation, and they are very close to each other. In Figure 10(c), the objectives  $f_1$  and  $f_4$  are highly traded off. At lower speeds, the robot consumes less power, and power consumption increases significantly with higher speeds. The Type I and Type II robust fronts have reduced their ranges for both objectives, and one end coincides with the  $\mathcal{PF}^{\text{org}}$ . This alignment suggests that there are a few robust points at faster speeds than at slower speeds. Additionally,  $f_1$  is found to be more sensitive than  $f_4$ . At the top-left of the scatter plot, there are points that take the highest possible speed to complete the gait cycle but with much higher power consumption. These solutions, enclosed in a rectangle, signify a high amount of power consumption at the higher speeds but without any robust solutions. There are no robust points near this region, as these solutions are responsible for the higher stability, but fulfilling these objectives is challenging due to the risk of small perturbations in the gait parameters causing constraint violations. This shows an absence of robust points at very high and very low speeds. The Type II robustness can provide much better points than Type I robustness in terms of convergence.



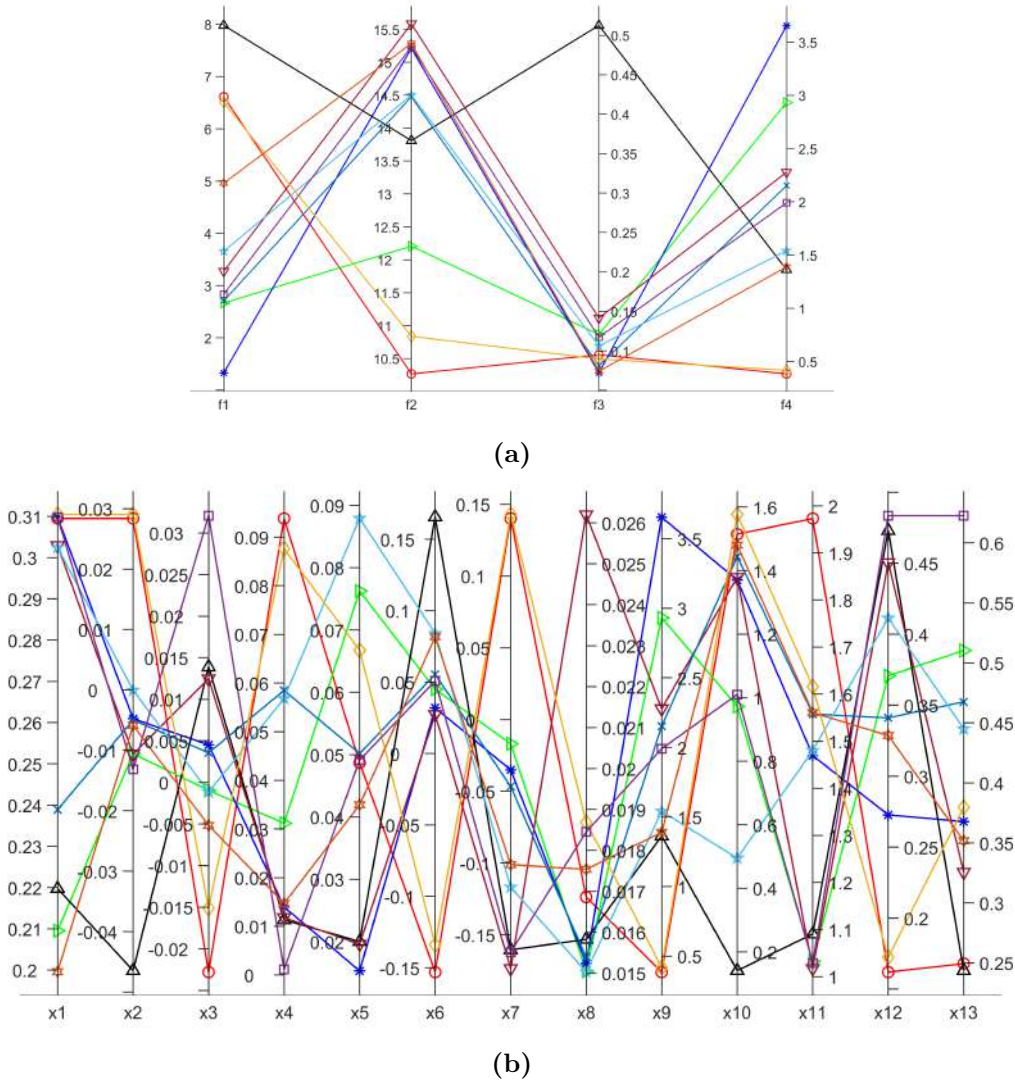
**Fig. 10.** Comparing original Pareto front with Type-I and II robust front by taking two objectives at a time during SSP.

In Figure 10(d), the scatter plot for  $f_2$  and  $f_3$  shows that there are no robust fronts for higher

stability with low torque fluctuations, as indicated by the enclosed circle. However, there are a few Type II robust points that can offer much higher stability than the rest of the robust fronts, enclosed in rectangles. In Figure 10(e), the scatter plot lying between  $f_2$  and  $f_4$  shows that stability increases at higher speeds. At the bottom-left, a few solutions enclosed in a rectangle demonstrate that a combination of gait parameters at higher speeds can achieve the higher stability, but these points are not robust. There are a few points enclosed in a rectangle from Type II robustness that can offer the better robust solutions with better stability at moderate speeds compared to the entire robust front. In Figure 10(f), the scatter plots between  $f_3$  and  $f_4$  show that at each time stamp, there are gait patterns that can offer similar torque fluctuations. The entire Type I and Type II robust fronts shift upwards, indicating the higher torque fluctuations at similar speeds, when compared with the original front. However, these points are more robust than the original front.

### 7.1.1. Pareto-Optimal Properties

To investigate the impact of various gait parameters on all four objectives, the properties of PO solutions have been explored. Ten random points are selected after combining the original, Type I, and Type II fronts are obtained using NSGA-III for analysis (Fig. 11a). Fig. 11b illustrates the search space and demonstrates how different gait parameters can affect all the objectives.



**Fig. 11.** (a) *Objective Space*: Ten random points are selected from the obtained Pareto-front using NSGA-III algorithm. (b) *Decision Space*: gait parameters (from  $x_1$  to  $x_{13}$ ) of the selected ten points belonging to the corresponding value of Pareto-front for the SSP problem.

The first three decision variables determined the hip trajectory. A higher hip height can reduce power consumption by requiring less torque from the hip and knee joints. Moreover, a positive slope for the hip trajectory (having a higher positive initial velocity to take the hip upward) can further decrease the joint angle variation in the ankle and knee joints. Subsequently, the final negative velocity brings the hip height lower to meet the starting hip height. There are more hip trajectories that can follow a straight path, resulting in low power consumption. Conversely, a lower hip height enhanced the stability margin, as it can keep the robot’s center of mass closer to the ground. Interestingly, a few solutions can provide better stability even with a high hip height, when the robot is moving at a faster pace. During the transition from the DSP to SSP, the initial and final sagittal velocities in the SSP ranged from 0.01 to 0.09 m/s. Relatively, the final sagittal velocities are found to be slightly higher. This happened because when the robot transitions back from the SSP to the DSP, going into the most stable phase, it can move at the higher velocities. This observation implies that the higher speeds can be attained during the DSP midway, whereas the SSP requires minimum velocity in its mid-phase for the increased stability and reduced power consumption. Nevertheless, only 50% of the lower and upper bounds of the velocities are utilized, as higher acceleration adversely affects the stability and power consumption due to the sudden torque requirements. The initial and final lateral velocities during the SSP predominantly have positive and negative values, facilitating sinusoidal hip movement. However, there are a few solutions that initially have negative values, then positive values, resulting in higher stability at faster paces. A lower swing height reduces power consumption, but one solution effectively has the higher swing height at lower power consumption and high torque fluctuation, when the robot is moving at a moderate speed. This solution can be effective during obstacle negotiation, when power consumption needs to be reduced.

Nonetheless, a few solutions effectively maintained good stability, low torque fluctuations with reduced power consumption, even at comparatively higher swing heights, during the robot’s movement from slow to moderate speed by utilizing arm movements. This shows that arm movements have an effect on stability, and moving these joints instead of the whole torso to maintain balance can reduce torque fluctuations. These solutions can also be used to navigate obstacles without excessive energy consumption. The last four decision parameters can control the movement of arms. Particularly, the  $\psi^{\mathcal{ER}}$  is found to move significantly more than the  $\psi^{SP}$  to improve stability, because it increased stability but with the slightly lesser power consumption. The higher power consumption in the  $\psi^{SP}$  can be attributed to its involvement in carrying the weight of the lower arm. Furthermore, when stability is primarily optimized, arm movements can play a crucial role in achieving the higher stability, as observable in Table 2, which shows the increased arm movement and variations in the  $\psi^{SP}$  and  $\psi^{\mathcal{ER}}$  joint angles.

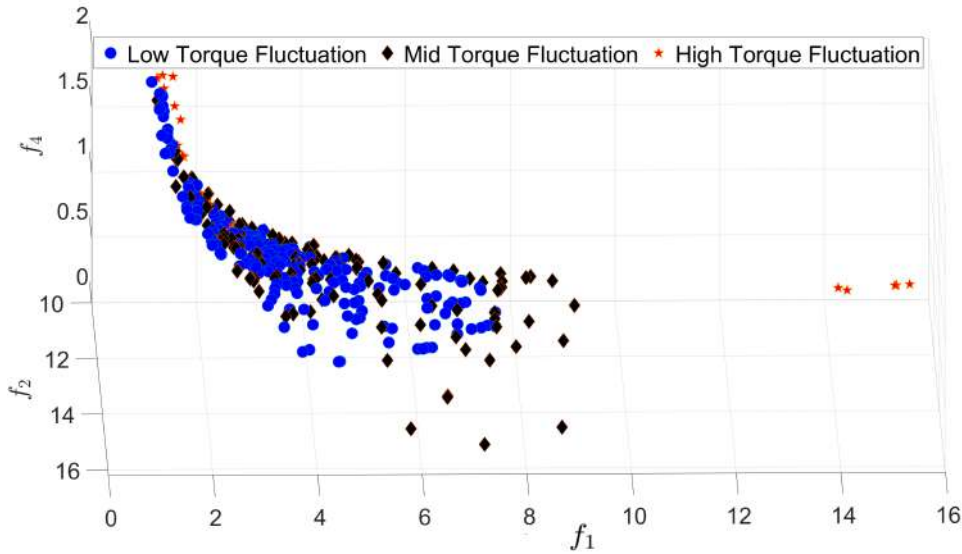
The discussion revealed a complex relationship between all objectives arising from variations in the gait parameters. These solutions have significantly different hip trajectories, velocity profiles, swing heights, and arm movements, indicating the complexity of humanoid robot bipedal locomotion, where slight changes in trajectory can have substantial impacts on the objectives. The selection of gait parameters depends on the specific application in which the robot is utilized. Gait parameters can be adjusted to prioritize either of the objectives, depending on the specific requirements of the application.

## 7.2. Double Support Phase

Figure 12 presents a scatter plot similar to SSP that maps the objective space during the DSP. The plot visualizes three objectives:  $f_1$ ,  $f_2$ , and  $f_4$ , plotted along the  $x$ ,  $y$ , and  $z$ -axes, respectively. Torque fluctuation ( $f_3$ ) is categorized into three distinct groups—low, medium, and high, and are represented by circle, diamond, and star markers, respectively.

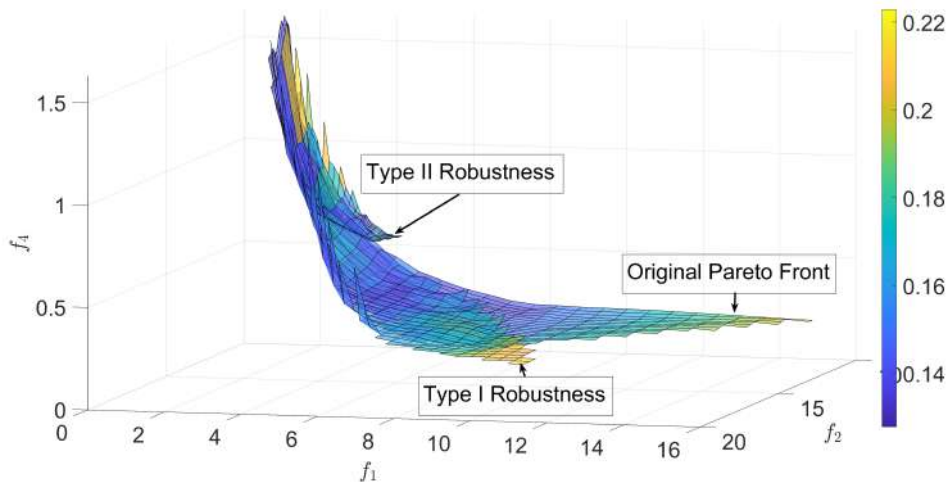
The scatter plot reveals a concentration of points that can exhibit high stability coupled with low power consumption, characterized by both low and high torque fluctuations at the slower speeds. Notably, on the far right of the plot, a select few solutions can demonstrate fast locomotion with enhanced stability, though at the cost of high torque fluctuations and power consumption. The distribution of the PF contract towards regions of high stability and low power consumption suggests a limited number of such solutions. However, as the  $f_4$  decreases, the robot presents a broader

spectrum of options at the higher speeds. These include selections between low to mid-range torque fluctuations and varying degrees of stability at the cost of higher power consumption. The choice among these solutions is contingent upon specific operational requirements, whether the goal is to achieve the faster movement, increased stability, or a more energy-efficient gait cycle.



**Fig. 12.** Scatter plot of original Pareto front during DSP.

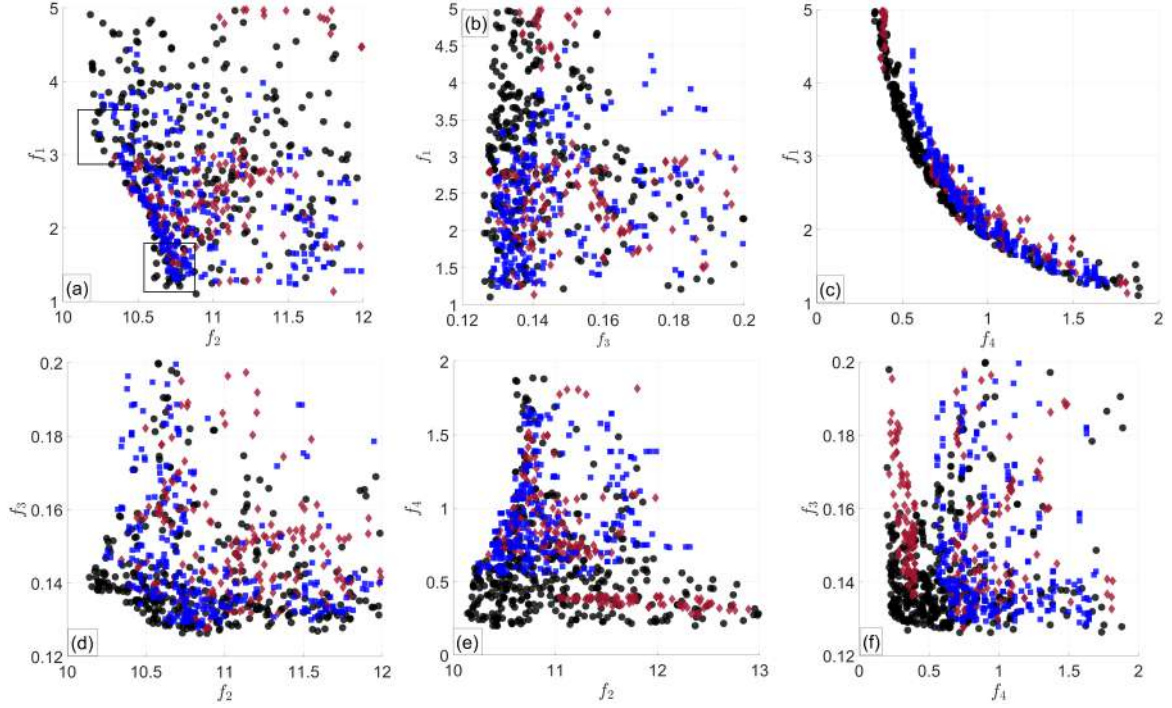
The surface plot for the DSP in Figure 13 reveals a different pattern of trade-offs between the objectives. The  $\mathcal{PF}^{\text{org}}$  covers a broad range of solutions, especially in terms of gait cycle time, demonstrating a clear trade-off between power consumption and stability. Type I robustness expands into regions not covered by the original front and Type II robustness, particularly in finding solutions with faster paces having lower gait cycle times. It offers diverse robust solutions. However, these solutions come at the expense of potential constraint violations, which could limit their practical applicability. Type II robustness, on the other hand, is more constrained in the objective space compared to Type I. It's unable to find robust solutions for very high speeds (low gait cycle time), but the solutions it does provide are generally more optimal and feasible than those of Type I. Type II concentrates on balancing power consumption and stability while maintaining low torque fluctuations. Type II robustness provides the more reliable and implementable solution as compared to Type I robustness during the DSP.



**Fig. 13.** Surface plot comparing the original Pareto-front with Type I and Type II robust solutions for the double support phase.

Figure 14 provides a series of bi-objective scatter plots for the DSP, where each plot pairs two

objectives to illustrate their interplay. The  $\mathcal{PF}^{\text{org}}$  is denoted by circle markers, Type I robust front is marked by diamond markers, and Type II robust front is indicated by square markers. Fig. 14(a) reveals the relationship between  $f_1$  and  $f_2$  in DSP. Notably, DSP features two distinct KR regions, highlighted within an enclosed rectangle. The second KR is particularly populated with Type II robust points, more so than the first. A slight deviation from the  $\mathcal{PF}^{\text{org}}$  is observed near the first KR, a contrast to the SSP scenario. Both robust fronts are closely aligned with the  $\mathcal{PF}^{\text{org}}$ , suggesting that robustness is more readily achieved due to the larger support polygon, allowing for the varied levels of power consumption and torque fluctuations, while maintaining desired stability. The formation of two KR regions in DSP can be attributed to two different types of motion that are possible during the walking cycle. In one trajectory, the robot walks at a faster pace, enhancing stability while consuming the higher amount of power. In the other trajectory, the robot walks at a slower pace, resulting in lower power consumption but comparatively less stability than the former trajectory.

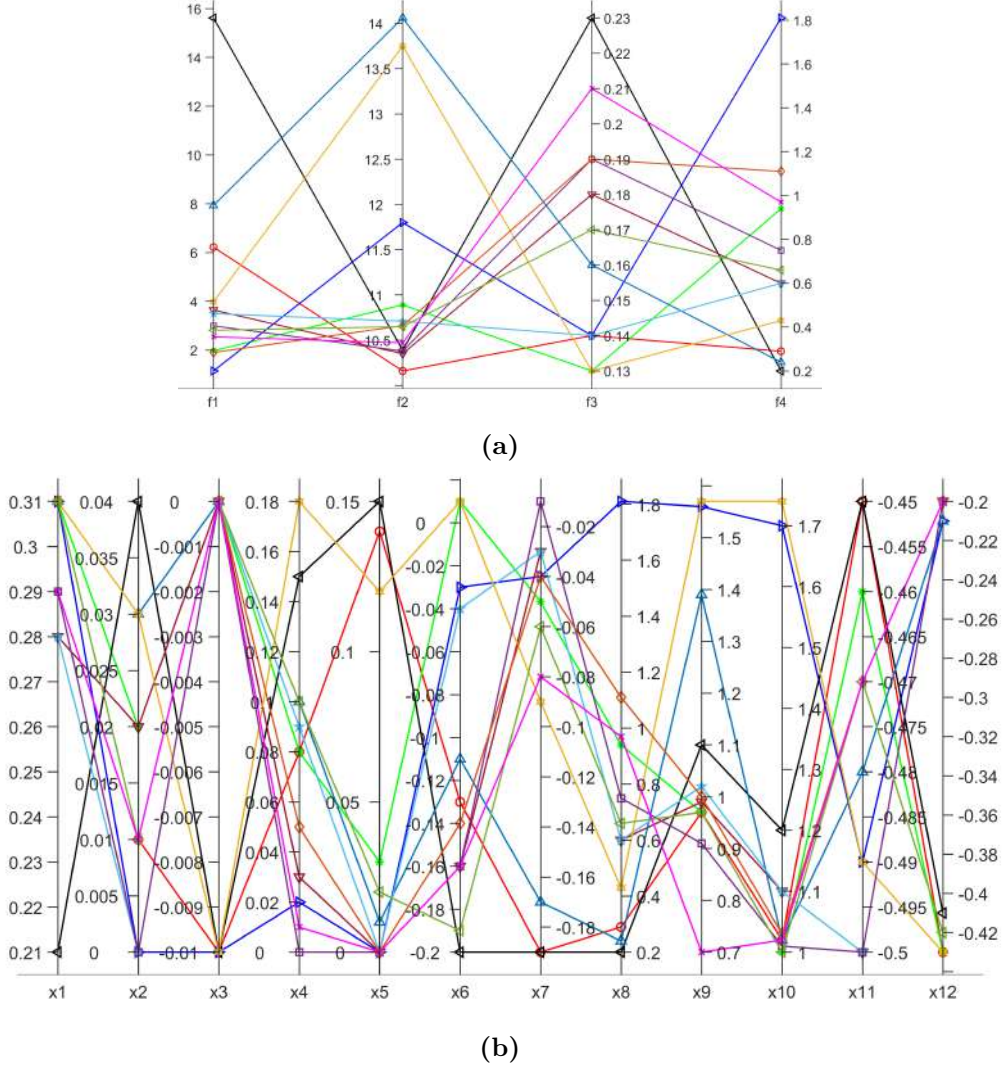


**Fig. 14.** Comparing original Pareto-front with Type-I and II robust front by taking two objectives at a time during DSP.

Fig. 14(b) shows the plot between  $f_1$  and  $f_3$ , indicating minimal values for similar gait parameters. The robust fronts, while maintaining a slight offset, lie in close proximity to the  $\mathcal{PF}^{\text{org}}$ , slightly shifting rightward in the objective space. Type I and II robust fronts are nearly overlapping. Fig. 14(c) presents the scatter plot showing a significant trade-off between minimizing power consumption and reducing gait cycle time. At lower speeds, the robot consumes less power, but as speed increases, power consumption escalates. The robust fronts exhibit lower values in the  $f_1$  objective for reduced time, with one end aligning with the  $\mathcal{PF}^{\text{org}}$ , indicating a prevalence of robust solutions at the slower speeds. The Type II robust front, however, presents a few robust solutions even at the faster locomotion speeds. Fig. 14(d) shows the relationship between  $f_2$  and  $f_3$  is evident, with higher stability correlating with increased torque fluctuations. Conversely, reducing torque fluctuations results in a compromise on stability. Fig. 14(e) depicts that higher speeds correlate with an increase in stability. The robust fronts have shifted towards the top right of the objective space, indicating a balance between robustness and optimality. In Fig. 14(f), different gait patterns, corresponding to each time stamp, offer varying torque fluctuations. The Type I and II robust fronts have moved upward and leftward, suggesting higher torque fluctuations at faster speeds compared to the original front, yet these points remain robust.

### 7.2.1. Pareto Optimal Properties

To elucidate the influence of gait parameters on the objectives, twelve random points are selected from a combination of all PF-original, Type I, and Type II robust fronts. Figure 15a visualizes the objective space in a PCP, focusing on objectives  $f_1$  to  $f_4$  during the DSP.



**Fig. 15.** (a) *Objective Space*: Twelve random points are selected from the combined PFs using the NSGA-III algorithm. (b) *Decision Space*: mapping of selected twelve points in decision variables (from  $x_1$  to  $x_{12}$ ) to its corresponding value in objective space.

For a comprehensive analysis, Figure 15b explores the search space of gait parameters, delineating their effects on all four objectives. In DSP, as observed in SSP, a greater hip height contributes to reduced power consumption by decreasing the joint angle variations at the knee and hip joints, thus necessitating less power for these joints. The hip trajectories identified are predominantly straight or exhibit a positive slope, with most trajectories showing minimal variation in maintaining a consistent hip height. In contrast, a lower hip height enhances stability by bringing the center of mass closer to the ground. The initial sagittal velocities during DSP are significantly higher than those during SSP, reflecting DSP's inherently stable nature. Conversely, the final sagittal velocities are lower, facilitating a smooth transition to the next SSP phase with minimal jerk. The DSP phase involves a lateral shift of the torso from the left leg to the right leg, moving in the negative direction along the y-axis. The initial lateral velocities span from  $-0.2$  to  $0.005$   $m/s$ , while the final lateral velocities range from  $-0.2$  to  $-0.02$   $m/s$ . Arm movements are generally less pronounced in DSP, with a few exceptions, as most of the motion in DSP occurs in the lateral direction, whereas arm movements are primarily in the sagittal plane. The  $\psi^{SP}$  exhibits more activity during DSP compared to the  $\psi^{ER}$ , which differs from

the patterns seen in SSP.

## 8. Concluding Remarks

This study has employed the NSGA-III algorithm to address the gait generation problem separately for the SSP and DSP of a humanoid robot’s gait cycle. The critical aspect lies in their implementation, as even slight perturbations may lead to deviations from the intended results. To enhance robustness, Type I and Type II approaches have been utilized. Type I considers the mean effective objective functions, while Type II introduces an inequality constraint on the normalized difference between the original and mean of perturbed objective functions. The present analysis has consistently favored Type II robustness concept, which outperforms the Type I robustness concept in terms of arriving at better robust solution quality, spread, and convergence of non-dominated solutions. Additionally, the Type II concept has proved more effective than Type I in addressing real-world complex problems with several functional constraints, suggesting its preference for similar robotic applications to find robust solutions in the presence of multiple constraints and variables.

The investigation of optimal gait solutions has highlighted significant trade-offs between the considered objectives – power consumption, stability, torque fluctuation, and gait cycle time. Notably, two distinct KRs have been identified during the SSP and DSP, signifying solutions that effectively balance the objectives and offer DMs a focused and viable set of options. While Type I and Type II are not able to find robust solutions near the first KR during the SSP, Type II has identified a few solutions for the first KR during the DSP, enhancing stability at high speeds. Furthermore, the analysis of gait parameters has revealed valuable insights. A higher hip height is associated with reduced power consumption and low torque fluctuations due to minimized hip and knee joint torque requirements, while a lower hip height contributes to increased stability with medium to high torque fluctuations by keeping the center of mass closer to the ground. Walking speed also has a direct impact, with slower speeds leading to lower power consumption and faster speeds enhancing stability with higher power consumption and moderate torque fluctuations. Hip height trajectories are predominantly observed to be straight during both the SSP and DSP, with a few solutions showing a distinctive pattern where the hip height initially increased and then gradually decreased. The optimization process has mostly utilized only a fraction of the available velocity range, as sudden acceleration can negatively impact stability and power consumption. Incorporating arm movements during the SSP to achieve higher swing height can enhance stability while minimizing power consumption, although their role becomes less significant during the DSP due to lateral movement.

The study has also suggested a Type II robust metric that can effectively handle constraints. It has shown promising results on the NAO humanoid robot’s real-world robotics gait generation problems, where Type II outperforms Type I due to its better handling of constraints. Future work could explore methods to compare different modified versions with the Type II robustness of other EMaO algorithms against NSGA-III, evaluating their effectiveness in solving other complex MaOPs. As only 16 solutions were generated due to the higher computational cost of finding the robust front, approaches to create robust solutions without significantly increasing the computational complexity of Evolutionary Algorithms should be developed. Identifying regions where robust solutions violate constraints may help reduce the trade-off between robustness and optimality. Direct identification of KRs could eliminate the need to exhaustively map the entire PF, streamlining the decision-making process by providing a reduced set of efficient, robust solutions that balance all objectives. Nevertheless, this study has highlighted the advantages of using Type II robustness and offered valuable recommendations regarding gait parameter selection in humanoid robots, emphasizing the importance of assessing robust solutions.

## Disclosure statement

The authors declare that they have no known competing interests that could have influenced the work reported in this article.

## Acknowledgements

The first author gratefully acknowledges the financial support of the Ministry of Education, Government of India.

## Data Availability Statement

The Kinematic data used in this study are publicly available on [http://doc.aldebaran.com/2-8/family/nao\\_technical/kinematics\\_naov6.html](http://doc.aldebaran.com/2-8/family/nao_technical/kinematics_naov6.html). The MATLAB code for all the algorithms used in this study are also freely available on <https://github.com/BIMK/PlatEMO>. The Data specific to DH parameter setting, inverse kinematics solutions, and any other file will be made available on reasonable request.

## References

- [1] S. Panagou, W. P. Neumann, and F. Fruggiero, “A scoping review of human robot interaction research towards industry 5.0 human-centric workplaces,” *International Journal of Production Research*, vol. 62, no. 3, pp. 974–990, 2024.
- [2] R. Rajendra and D. K. Pratihari, “Analysis of double support phase of biped robot and multi-objective optimization using genetic algorithm and particle swarm optimization algorithm,” *Śadhanā*, vol. 40, pp. 549–575, 2015.
- [3] M. Raj, V. B. Semwal, and G. C. Nandi, “Multiobjective optimized bipedal locomotion,” *International Journal of Machine Learning and Cybernetics*, vol. 10, pp. 1997–2013, 2019.
- [4] M. J. Mahmoodabadi and M. Taherkhorsandi, “Intelligent control of biped robots: optimal fuzzy tracking control via multi-objective particle swarm optimization and genetic algorithms,” *AUT Journal of Mechanical Engineering*, vol. 4, no. 2, pp. 183–192, 2020.
- [5] R. Abedzadeh Maafi, S. Etemadi Haghghi, and M. J. Mahmoodabadi, “Pareto optimal design of a fuzzy adaptive sliding mode controller for a three-link model of a biped robot via the multi-objective improved team game algorithm,” *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, vol. 44, no. 9, p. 428, 2022.
- [6] P. Fernandes, R. Oliveira, and J. Fonseca Neto, “Trajectory planning of autonomous mobile robots applying a particle swarm optimization algorithm with peaks of diversity,” *Applied Soft Computing*, vol. 116, p. 108108, 2022.
- [7] F. H. Ajeil, I. K. Ibraheem, M. A. Sahib, and A. J. Humaidi, “Multi-objective path planning of an autonomous mobile robot using hybrid PSO-MFB optimization algorithm,” *Applied Soft Computing*, vol. 89, p. 106076, 2020.
- [8] T. T. Mac, C. Copot, D. T. Tran, and R. D. Keyser, “A hierarchical global path planning approach for mobile robots based on multi-objective particle swarm optimization,” *Applied Soft Computing*, vol. 59, pp. 68–76, 2017.
- [9] P. Kulvanit, N. Chaiyaratana, and D. Laowattana, “Biped fast walking gait shaping via evolutionary multi-objective optimization,” in *2007 IEEE Congress on Evolutionary Computation*, pp. 4019–4026, IEEE, 2007.
- [10] D. Gong, J. Yan, and G. Zuo, “A review of gait optimization based on evolutionary computation,” *Applied Computational Intelligence and Soft Computing*, vol. 2010, 2010.
- [11] Y. Uno, M. Kawato, and R. Suzuki, “Formation and control of optimal trajectory in human multijoint arm movement,” *Biological Cybernetics 1989 61:2*, vol. 61, pp. 89–101, 6 1989.

- [12] C. Niehaus, T. Röfer, and T. Laue, “Gait optimization on a humanoid robot using particle swarm optimization,” in *Proceedings of the Second Workshop on Humanoid Soccer Robots in conjunction with the*, pp. 1–7, Nov 2007.
- [13] V.-H. Dau, C.-M. Chew, and A.-N. Poo, “Achieving energy-efficient bipedal walking trajectory through GA-based optimization of key parameters,” *International Journal of Humanoid Robotics*, vol. 6, pp. 609–629, 2009.
- [14] Li-Yang Wang, Z. Liu, Xiao-Jie Zeng, and Y. Zhang, “Gait control of humanoid robots via fuzzy logic and iterative optimization,” in *Proceedings of the 30th Chinese Control Conference*, pp. 3931–3936, IEEE, July 2011.
- [15] L. Gong, R. Zhao, J. Liang, L. Li, M. Zhu, Y. Xu, X. Tai, X. Qiu, H. He, F. Guo, *et al.*, “Generation of walking motions for the biped ascending slopes based on genetic algorithm,” in *International Conference on Swarm Intelligence*, pp. 201–209, Springer, June 2018.
- [16] Vikas, D. R. Parhi, and A. K. Kashyap, “Humanoid robot path planning using memory-based gravity search algorithm and enhanced differential evolution approach in a complex environment,” *Expert Systems with Applications*, vol. 215, p. 119423, 2023.
- [17] P. Gupta, D. K. Pratihar, and K. Deb, “A knee-based multi-objective optimization for gait cycle of 25-DOF NAO humanoid robot,” in *International conference on soft computing for problem-solving*, pp. 47–62, Springer, 2023.
- [18] X. Leng, S. Piao, L. Chang, Z. He, and Z. Zhu, “Parameter design of biped robot motion system based on multi-objective optimization,” *Journal of Intelligent & Fuzzy Systems*, vol. 41, no. 3, pp. 4307–4318, 2021.
- [19] K. Sanprasisit, “Multi-objective optimization algorithm of humanoid robot walking on a narrow beam,” *International Journal of Mechanical Engineering and Robotics Research*, vol. 9, no. 12, pp. 1548–1559, 2020.
- [20] D. Yadav, P. Ramu, and K. Deb, “Finding robust solutions for many-objective optimization using NSGA-III,” in *2023 IEEE Congress on Evolutionary Computation (CEC)*, pp. 1–8, IEEE, 2023.
- [21] P. Gupta, D. K. Pratihar, and K. Deb, “Analysis and optimization of gait cycle of 25-dof NAO robot using particle swarm optimization and genetic algorithms,” *International Journal of Humanoid Robotics*, vol. 21, no. 02, pp. 2350011(1)–2350011(44), 2024.
- [22] A. K. Kashyap, D. R. Parhi, M. K. Muni, and K. K. Pandey, “A hybrid technique for path planning of humanoid robot NAO in static and dynamic terrains,” *Applied Soft Computing*, vol. 96, p. 106581, 2020.
- [23] P. B. Kumar, M. K. Muni, and D. R. Parhi, “Navigational analysis of multiple humanoids using a hybrid regression-fuzzy logic control approach in complex terrains,” *Applied Soft Computing*, vol. 89, p. 106088, 2020.
- [24] A. K. Kashyap and D. R. Parhi, “Implementation of intelligent navigational techniques for inter-collision avoidance of multiple humanoid robots in complex environment,” *Applied Soft Computing*, vol. 124, p. 109001, 2022.
- [25] M. K. Muni, D. R. Parhi, P. B. Kumar, C. Sahu, and S. Kumar, “Towards motion planning of humanoids using a fuzzy embedded neural network approach,” *Applied Soft Computing*, vol. 119, p. 108588, 2022.
- [26] Vikas and D. R. Parhi, “Chaos-based optimal path planning of humanoid robot using hybridized regression-gravity search algorithm in static and dynamic terrains,” *Applied Soft Computing*, vol. 140, p. 110236, 2023.

- [27] K. Deb and H. Jain, “An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part I: Solving problems with box constraints,” *IEEE Transactions on Evolutionary Computation*, vol. 18, no. 4, pp. 577–601, 2014.
- [28] H. Jain and K. Deb, “An evolutionary many-objective optimization algorithm using reference-point based nondominated sorting approach, part II: Handling constraints and extending to an adaptive approach,” *IEEE Transactions on evolutionary computation*, vol. 18, no. 4, pp. 602–622, 2013.
- [29] R. Jiao, S. Zeng, C. Li, S. Yang, and Y.-S. Ong, “Handling constrained many-objective optimization problems via problem transformation,” *IEEE Transactions on Cybernetics*, vol. 51, no. 10, pp. 4834–4847, 2020.
- [30] M. Mahmoodabadi, M. Taherkhorsandi, and A. Bagheri, “Optimal robust sliding mode tracking control of a biped robot based on ingenious multi-objective PSO,” *Neurocomputing*, vol. 124, pp. 194–209, 2014.
- [31] M. K. Muni, D. R. Parhi, and P. B. Kumar, “Improved motion planning of humanoid robots using bacterial foraging optimization,” *Robotica*, vol. 39, pp. 123–136, 2021.
- [32] A. K. Kashyap, D. R. Parhi, and A. Pandey, “Multi-objective optimization technique for trajectory planning of multi-humanoid robots in cluttered terrain,” *ISA transactions*, vol. 125, pp. 591–613, 2022.
- [33] C. Liu, A. Liu, R. Wang, H. Zhao, and Z. Lu, “Path planning algorithm for multi-locomotion robot based on multi-objective genetic algorithm with elitist strategy,” *Micromachines*, vol. 13, no. 4, p. 616, 2022.
- [34] M. Russo, S. Herrero, O. Altuzarra, and M. Ceccarelli, “Kinematic analysis and multi-objective optimization of a 3-UPR parallel mechanism for a robotic leg,” *Mechanism and Machine Theory*, vol. 120, pp. 192–202, 2018.
- [35] T. Kiyokawa, K. Harada, W. Wan, T. Ishikura, N. Miyaji, and G. Matsuda, “Many-objective-optimized semi-automated robotic disassembly sequences,” *arXiv preprint arXiv:2401.01817*, 2024.
- [36] N. Ito and H. Hasegawa, “Robust optimization under uncertain factors of environment for simple gait of biped robots,” in *Proceedings of the 6th EUROSIM Congress on Modelling and Simulation, 2007*, 2007.
- [37] C. Collette, A. Micaelli, C. Andriot, and P. Lemerle, “Robust balance optimization control of humanoid robots with multiple non coplanar grasps and frictional contacts,” in *2008 IEEE International Conference on Robotics and Automation*, pp. 3187–3193, IEEE, 2008.
- [38] T. Hayashi, M. Yamakita, Y. Hanazawa, and F. Asano, “Robust walking of biped robot on uneven terrain using effect of wobbling mass,” in *2014 IEEE International Conference on Robotics and Biomimetics (ROBIO 2014)*, pp. 2080–2085, IEEE, 2014.
- [39] I. Dakhli, E. Maherzi, and M. Besbes, “Robust walking control algorithm of biped robot in rough ground,” in *2016 13th International Multi-Conference on Systems, Signals & Devices (SSD)*, pp. 762–767, IEEE, 2016.
- [40] M. Missura, M. Bennewitz, and S. Behnke, “Capture steps: Robust walking for humanoid robots,” *International journal of humanoid robotics*, vol. 16, no. 06, p. 1950032, 2019.
- [41] M. H. Yeganegi, M. Khadiv, S. A. A. Moosavian, J.-J. Zhu, A. Del Prete, and L. Righetti, “Robust humanoid locomotion using trajectory optimization and sample-efficient learning,” in *2019 IEEE-RAS 19th International Conference on Humanoid Robots (Humanoids)*, pp. 170–177, IEEE, 2019.

- [42] M. H. Yeganegi, M. Khadiv, A. Del Prete, S. A. A. Moosavian, and L. Righetti, “Robust walking based on MPC with viability guarantees,” *IEEE Transactions on Robotics*, vol. 38, no. 4, pp. 2389–2404, 2021.
- [43] Z. Wang, W. Wei, R. Yu, J. Wu, and Q. Zhu, “Toward understanding key estimation in learning robust humanoid locomotion,” *arXiv preprint arXiv:2403.05868*, 2024.
- [44] S. Zhang, F. Ni, X. Li, X. Li, Y. Ji, and X. Shao, “Robustness and push recovery of humanoid robots using predictive and whole-body control,” in *2024 IEEE International Conference on Unmanned Systems (ICUS)*, pp. 871–877, IEEE, 2024.
- [45] Z. Zhao, H. Huang, S. Sun, C. Li, and W. Xu, “Fusing dynamics and reinforcement learning for control strategy: Achieving precise gait and high robustness in humanoid robot locomotion,” in *2024 IEEE-RAS 23rd International Conference on Humanoid Robots (Humanoids)*, pp. 1072–1079, IEEE, 2024.
- [46] S. Saeedvand, H. S. Aghdasi, and J. Baltes, “Robust multi-objective multi-humanoid robots task allocation based on novel hybrid metaheuristic algorithm,” *Applied Intelligence*, vol. 49, no. 12, pp. 4097–4127, 2019.
- [47] Y. Chang, K. Ebadi, C. E. Denniston, M. F. Ginting, A. Rosinol, A. Reinke, M. Palieri, J. Shi, A. Chatterjee, B. Morrell, *et al.*, “Lamp 2.0: A robust multi-robot slam system for operation in challenging large-scale underground environments,” *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 9175–9182, 2022.
- [48] G. Gaspard, F. Fabiani, M. Garabini, L. Pallottino, M. Catalano, G. Grioli, R. Persichin, and A. Bicchi, “Robust optimization of system compliance for physical interaction in uncertain scenarios,” in *2016 IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids)*, pp. 911–918, IEEE, 2016.
- [49] Y. Cheng, L. Sun, C. Liu, and M. Tomizuka, “Towards efficient human-robot collaboration with robust plan recognition and trajectory prediction,” *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 2602–2609, 2020.
- [50] J. Luo and K. Hauser, “Robust trajectory optimization under frictional contact with iterative learning,” *Autonomous Robots*, vol. 41, pp. 1447–1461, 2017.
- [51] L. Drnach and Y. Zhao, “Robust trajectory optimization over uncertain terrain with stochastic complementarity,” *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 1168–1175, 2021.
- [52] L. Drnach, J. Z. Zhang, and Y. Zhao, “Mediating between contact feasibility and robustness of trajectory optimization through chance complementarity constraints,” *Frontiers in Robotics and AI*, vol. 8, p. 785925, 2022.
- [53] M. Bogdanovic, M. Khadiv, and L. Righetti, “Model-free reinforcement learning for robust locomotion using demonstrations from trajectory optimization,” *Frontiers in Robotics and AI*, vol. 9, p. 854212, 2022.
- [54] Z. Sun, T. Heng, L. Zhao, S. Liu, Y. Lian, and K. Liu, “A trajectory tracking method based on robust model predictive control for a bionic ankle-foot aided by a tensegrity mechanism,” *Engineering Optimization*, vol. 56, no. 10, pp. 1660–1682, 2024.
- [55] M. Ramezani, G. Tinchev, E. Iuganov, and M. Fallon, “Online lidar-slam for legged robots with robust registration and deep-learned loop closure,” in *2020 IEEE international conference on robotics and automation (ICRA)*, pp. 4158–4164, IEEE, 2020.
- [56] A. E. Gkikakis and R. Featherstone, “Robust analysis for mechanism and behavior co-optimization of high-performance legged robots,” in *2022 IEEE-RAS 21st International Conference on Humanoid Robots (Humanoids)*, pp. 752–758, IEEE, 2022.

- [57] Z.-W. Luo, M. Ito, A. Kato, and K. Ito, “Nonlinear robust control for robot compliant manipulation on dynamic environments,” *Advanced robotics*, vol. 10, no. 2, pp. 213–227, 1995.
- [58] C.-P. Kuan and K.-y. Young, “Reinforcement learning and robust control for robot compliance tasks,” *Journal of Intelligent and Robotic Systems*, vol. 23, pp. 165–182, 1998.
- [59] S. Kim and J. Ryu, “Robust interaction control for environments having uncertainties,” *Robotics and Autonomous Systems*, vol. 151, p. 104023, 2022.
- [60] T. B. Jørgensen, A. Wolniakowski, H. G. Petersen, K. Debrabant, and N. Krüger, “Robust optimization with applications to design of context specific robot solutions,” *Robotics and Computer-Integrated Manufacturing*, vol. 53, pp. 162–177, 2018.
- [61] M. Charbonneau, V. Modugno, F. Nori, G. Oriolo, D. Pucci, and S. Ivaldi, “Learning robust task priorities of QP-based whole-body torque-controllers,” in *2018 IEEE-RAS 18th International Conference on Humanoid Robots (Humanoids)*, pp. 1–9, IEEE, 2018.
- [62] X. Luo, D. Xia, and C. Zhu, “Impact dynamics-based torso control for dynamic walking biped robots,” *International Journal of Humanoid Robotics*, vol. 15, no. 03, p. 1850004, 2018.
- [63] X. Mu and Q. Wu, “A complete dynamic model of five-link bipedal walking,” in *Proceedings of the 2003 American Control Conference, 2003.*, vol. 6, pp. 4926–4931, IEEE, 2003.
- [64] P. R. Vundavilli, S. K. Sahu, and D. K. Pratihar, “Online dynamically balanced ascending and descending gait generations of a biped robot using soft computing,” *International Journal of Humanoid Robotics*, vol. 4, pp. 777–814, 2007.
- [65] P. R. Vundavilli, S. K. Sahu, and D. K. Pratihar, “Dynamically balanced ascending and descending gaits of a two-legged robot,” *International Journal of Humanoid Robotics*, vol. 4, pp. 717–751, 2007.
- [66] Tushar, P. R. Vundavilli, and D. K. Pratihar, “Dynamically balanced ascending gait generation of a biped robot negotiating staircase,” in *2008 IEEE Region 10 and the Third international Conference on Industrial and Information Systems*, pp. 1–6, Dec 2008.
- [67] P. R. Vundavilli and D. K. Pratihar, “Soft computing-based gait planners for a dynamically balanced biped robot negotiating sloping surfaces,” *Applied Soft Computing*, vol. 9, no. 1, pp. 191–208, 2009.
- [68] P. R. Vundavilli and D. K. Pratihar, “Dynamically balanced optimal gaits of a ditch-crossing biped robot,” *Robotics and Autonomous Systems*, vol. 58, no. 4, pp. 349–361, 2010.
- [69] F. M. Silva and J. T. Machado, “Energy analysis during biped walking,” in *Proceedings 1999 IEEE International Conference on Robotics and Automation (Cat. No. 99CH36288C)*, vol. 1, pp. 59–64, IEEE, 1999.
- [70] P. H. Channon, S. H. Hopkins, and D. T. Pham, “A variational approach to the optimization of gait for a bipedal robot,” *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, vol. 210, pp. 177–184, 1996.
- [71] Y. Farzaneh, A. Akbarzadeh, and A. A. Akbari, “Online bio-inspired trajectory generation of seven-link biped robot based on T–S fuzzy system,” *Applied Soft Computing*, vol. 14, pp. 167–180, 2014. *Evolving Soft Computing Techniques and Applications*.
- [72] SoftBank Robotics Developer Center, “Kinematics data: Links, joints, and body frames,” 2021. [Online; accessed 15-August-2021].
- [73] SoftBank Robotics Developer Center, “The masses and COM positions for NAO,” 2021. [Online; accessed 15-August-2021].

- [74] J. Denavit and R. S. Hartenberg, “A kinematic notation for lower-pair mechanisms based on matrices,” *Journal of Applied Mechanics*, 1955.
- [75] N. Kofinas, E. Orfanoudakis, and M. G. Lagoudakis, “Complete analytical forward and inverse kinematics for the NAO humanoid robot,” *Journal of Intelligent & Robotic Systems*, vol. 77, no. 2, pp. 251–264, 2015.
- [76] E. Hashemi and M. Ghaffari Jadidi, “Dynamic modeling and control study of the NAO biped robot with improved trajectory planning,” *Materials with Complex Behaviour II: Properties, Non-Classical Materials and New Technologies*, pp. 671–688, 2012.
- [77] E. Hashemi and A. Khajepour, “Kinematic and three-dimensional dynamic modeling of a biped robot,” *Proceedings of the Institution of Mechanical Engineers, Part K: Journal of Multi-body Dynamics*, vol. 231, no. 1, pp. 57–73, 2017.
- [78] M. Alibeigi, S. Rabiee, and M. N. Ahmadabadi, “Inverse kinematics based human mimicking system using skeletal tracking technology,” *Journal of Intelligent & Robotic Systems*, vol. 85, pp. 27–45, 2017.
- [79] S. Sakka, L. P. Poubel, and D. Cehajic, “Tasks prioritization for whole-body realtime imitation of human motion by humanoid robots,” in *Digital Intelligence (DI2014)*, pp. 5–p, 2014.
- [80] F. Wang, C. Tang, Y. Ou, and Y. Xu, “A real-time human imitation system,” in *Proceedings of the 10th World Congress on Intelligent Control and Automation*, pp. 3692–3697, IEEE, 2012.
- [81] X. Lv, J. Chai, and S. Xia, “Data-driven inverse dynamics for human motion,” *ACM Transactions on Graphics*, vol. 35, Nov 2016.
- [82] C. Hernández-Santos, R. Soto, and E. Rodriguez, “Design and dynamic modeling of humanoid biped robot e-robot,” in *2011 IEEE Electronics, Robotics and Automotive Mechanics Conference*, pp. 191–196, Nov 2011.
- [83] M. Rameez and L. A. Khan, “Modeling and dynamic analysis of the biped robot,” in *2015 15th International Conference on Control, Automation and Systems (ICCAS)*, pp. 1149–1153, Oct 2015.
- [84] A. K. Kashyap, D. R. Parhi, and S. Kumar, “Dynamic stabilization of NAO humanoid robot based on whole-body control with simulated annealing,” *International Journal of Humanoid Robotics*, vol. 17, no. 03, p. 2050014, 2020.
- [85] X. Bajrami, A. Dermaku, A. Shala, and R. Likaj, “Kinematics and dynamics modelling of the biped robot,” *IFAC Proceedings Volumes (IFAC-PapersOnline)*, vol. 15, no. PART 1, pp. 69–73, 2013.
- [86] M. Folgheraiter and B. Aubakir, “Design and modeling of a lightweight and low power consumption full-scale biped robot,” *International Journal of Humanoid Robotics*, vol. 15, Oct 2018.
- [87] M. Khadiv, M. Ezati, and S. A. A. Moosavian, “A computationally efficient inverse dynamics solution based on virtual work principle for biped robots,” *Iranian Journal of Science and Technology, Transactions of Mechanical Engineering*, vol. 43, no. 1, pp. 37–52, 2019.
- [88] D. K. Pratihar, *Fundamentals of Robotics*. Narosa Publishing House Pvt. Ltd., 2017.
- [89] K. S. Fu, R. C. Gonzalez, and C. S. Lee, *Robotics: Control, Sensing, Vision and Intelligence*. Tata McGraw-Hill Education, 1987.
- [90] M. Vukobratović and B. Borovac, “Zero-moment point—thirty five years of its life,” *International journal of humanoid robotics*, vol. 1, pp. 157–173, 2004.
- [91] “NAO Version 6 description.” <https://www.aldebaran.com/en/nao>. Accessed: 2023-06-22.

- [92] Y. J. Seo and Y. S. Yoon, “Design of a robust dynamic gait of the biped using the concept of dynamic stability margin,” *Robotica*, vol. 13, no. 5, p. 461–468, 1995.
- [93] S. Kajita, H. Hirukawa, K. Harada, and K. Yokoi, *Introduction to Humanoid Robotics*, vol. 101. Springer, 2014.
- [94] J. Nishii, K. Ogawa, and R. Suzuki, “The optimal gait pattern in hexapods based on energetic efficiency,” in *Proceedings of the 3rd international symposium on artificial life and robotics*, pp. 106–109, Citeseer, 1998.
- [95] 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.
- [96] C. A. C. Coello, *Evolutionary algorithms for solving multi-objective problems*. Springer, 2007.
- [97] J. Branke, “Creating robust solutions by means of evolutionary algorithms,” in *Parallel Problem Solving from Nature—PPSN V: 5th International Conference Amsterdam, The Netherlands September 27–30, 1998 Proceedings 5*, pp. 119–128, Springer, 1998.
- [98] J. Branke, “Efficient evolutionary algorithms for searching robust solutions,” in *Evolutionary Design and Manufacture: Selected Papers from ACDM’00*, pp. 275–285, Springer, 2000.
- [99] Y. Jin and B. Sendhoff, “Trade-off between performance and robustness: An evolutionary multiobjective approach,” in *International conference on evolutionary multi-criterion optimization*, pp. 237–251, Springer, 2003.
- [100] S. Tsutsui and A. Ghosh, “Genetic algorithms with a robust solution searching scheme,” *IEEE transactions on Evolutionary Computation*, vol. 1, no. 3, pp. 201–208, 1997.
- [101] E. J. Hughes, “Evolutionary multi-objective ranking with uncertainty and noise,” in *International Conference on Evolutionary Multi-Criterion Optimization*, pp. 329–343, Springer, 2001.
- [102] K. Deb and H. Gupta, “Searching for robust pareto-optimal solutions in multi-objective optimization,” in *International conference on evolutionary multi-criterion optimization*, pp. 150–164, Springer, 2005.
- [103] 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.
- [104] H. Gupta and K. Deb, “Handling constraints in robust multi-objective optimization,” in *2005 IEEE Congress on Evolutionary Computation*, vol. 1, pp. 25–32, IEEE, 2005.
- [105] Q. Zhou, P. Jiang, X. Huang, F. Zhang, and T. Zhou, “A multi-objective robust optimization approach based on gaussian process model,” *Structural and Multidisciplinary Optimization*, vol. 57, pp. 213–233, 2018.
- [106] S. Mirjalili and A. Lewis, “Novel frameworks for creating robust multi-objective benchmark problems,” *Information Sciences*, vol. 300, pp. 158–192, 2015.
- [107] S. Mirjalili and A. Lewis, “Hindrances for robust multi-objective test problems,” *Applied Soft Computing*, vol. 35, pp. 333–348, 2015.
- [108] S. Mirjalili and A. Lewis, “Novel performance metrics for robust multi-objective optimization algorithms,” *Swarm and Evolutionary Computation*, vol. 21, pp. 1–23, 2015.
- [109] Q. Zhang and H. Li, “Moea/d: A multiobjective evolutionary algorithm based on decomposition,” *IEEE Transactions on evolutionary computation*, vol. 11, no. 6, pp. 712–731, 2007.

- [110] E. Zitzler, M. Laumanns, and L. Thiele, “Spea2: Improving the strength pareto evolutionary algorithm,” *TIK report*, vol. 103, 2001.
- [111] J. Bader and E. Zitzler, “Hype: An algorithm for fast hypervolume-based many-objective optimization,” *Evolutionary computation*, vol. 19, no. 1, pp. 45–76, 2011.
- [112] Y. Tian, R. Cheng, X. Zhang, and Y. Jin, “PlatEMO: A MATLAB platform for evolutionary multi-objective optimization,” *IEEE Computational Intelligence Magazine*, vol. 12, no. 4, pp. 73–87, 2017.
- [113] C. M. Fonseca, L. Paquete, and M. López-Ibáñez, “An improved dimension-sweep algorithm for the hypervolume indicator,” in *2006 IEEE international conference on evolutionary computation*, pp. 1157–1163, IEEE, 2006.
- [114] J. Blank and K. Deb, “Pymoo: Multi-objective optimization in python,” *IEEE Access*, vol. 8, pp. 89497–89509, 2020.
- [115] S. Lee, J. Yosinski, K. Glette, H. Lipson, and J. Clune, “Evolving gaits for physical robots with the hyperneat generative encoding: The benefits of simulation,” in *Applications of Evolutionary Computation: 16th European Conference, EvoApplications 2013, Vienna, Austria, April 3-5, 2013. Proceedings 16*, pp. 540–549, Springer, 2013.
- [116] C. Paul, J. W. Roberts, H. Lipson, and F. V. Cuevas, “Gait production in a tensegrity based robot,” in *ICAR’05. Proceedings., 12th International Conference on Advanced Robotics, 2005.*, pp. 216–222, IEEE, 2005.
- [117] T. Takagi, K. Takadama, and H. Sato, “Incremental lattice design of weight vector set,” in *Proceedings of the 2020 Genetic and Evolutionary Computation Conference Companion*, pp. 1486–1494, 2020.