2024 10th International Conference on Control, Decision and Information Technologies
CoDIT 2024 | Valletta, Malta / July 01-04, 2024 Technically co-sponsored by IEEE & IFAC

2024 10th International Conference on Control, Decision and Information Technologies (CoDIT) | 979-8-3503-7397-4/24/$31.00 ©2024 IEEE | DOI: 10.1109/CoDIT62066.2024.10708340

Stabilization of Driving Velocity Constraints for Self-balanced Robot

Ibrahim Abdel-Hady[1], Mona A. Bayoumi[1], Nader A. Mansour[1,2] and Ayman A. Nada [3]

***Abstract*— Non-holonomic constraints impose restrictions on the allowable velocities or motions of the system. These constraints may arise from physical interactions or mechanical limitations. Stabilizing constraints in a non-holonomic multibody system often involves employing numerical methods due to the complexity of the constraints and the dynamic nature of the system. Moreover, for self-balancing robots, driving constraints may be introduced to the system in the velocity level, and proper management of these constraints is crucial in the design and analysis of mechanisms, vehicles, robotics, and other complex systems. In this paper, we present an approach for stabilizing the driving velocity constraints, along with other holonomic and non-holonomic constraints, of a self-balanced robot. The proposed approach is intended for use in the numerical integration process of the Differential Algebraic Equations of multibody system dynamics, and not for real-time control. Successful numerical integration enables the calculation of driving forces in an open-loop manner. The paper proves that fuzzy logic control can be utilized effectively for driving constraints stabilization at the velocity level.**

I. INTRODUCTION

Unlike holonomic systems, where constraints can be fully described by equations involving positions, velocities, and accelerations, non-holonomic constraints are typically inequalities involving velocities and cannot be integrated directly to yield position constraints. These constraints often arise due to the nature of the robot's mechanical design, such as wheeled robots or vehicles [1], [2]. Stabilizing constraints in a non-holonomic multibody system is more challenging than in holonomic systems. This is because non-holonomic constraints limit the possible motions of the system without completely defining them. [3]. numerical methods that commonly used for constraint stabilization in non-holonomic multibody systems include [4]:

- Holonomic Approximation: Approximate non-holonomic constraints with holonomic constraints that can be integrated more easily. This allows to treat the system as a holonomic multibody system, simplifying the constraint stabilization problem.
- Projection Methods: This method involves integrating the system's equations of motion subject to the constraints using explicit numerical integrators, followed by a projection step to enforce the constraints onto the constraint manifold at each time step. Projection methods ensure that the system remains close to the constraint manifold throughout the simulation.
- Penalty Methods: Penalize violations of the constraints in the system's objective function during numerical optimization or simulation. The penalty terms increase as the constraints are violated, effectively pushing the system towards satisfying the constraints.

A self-balancing robot can indeed be modeled as a non-holonomic multibody system due to the constraints it faces. In this context, the non-holonomic constraints typically arise from the fact that the robot's motion is restricted by its wheels' kinematics and the need to maintain balance.
Self-balancing robots are versatile due to their dynamic stabilization capabilities, adaptability, and compact design. Their applications include personal transportation, assistive devices, warehouse management, surveillance, telepresence, and entertainment [5], [6].

II. MULTIBODY MODEL OF SELF-BALANCED ROBOT

A. System Description

The Self-Balanced robot structure, is shown in Fig.(2), which is consists of the following components:

- Pendulum: This is the vertical component, upwards from the base where the wheels are attached. It acts as an inverted pendulum, which is a common setup for self-balancing robots. The pendulum's position is typically monitored by sensors to help the robot maintain its balance.
- Wheels: There are two wheels labeled 'left' and 'right'. These wheels are critical for the robot's movement and balance, and are most likely powered by motors. Each motor can be controlled independently, allowing the robot to manoeuvre and remain upright.

The generalized coordinates vector of the system can be written as $\mathbf{q}^T = \begin{bmatrix} \mathbf{q}^{1^T} & \mathbf{q}^{2^T} & \mathbf{q}^{3^T} \end{bmatrix}^T$, where, $\mathbf{q}^i = \begin{bmatrix} \mathbf{R}^{i^T} & \theta^{i^T} \end{bmatrix}^T = \begin{bmatrix} R^i_x & R^i_y & R^i_z & \phi^i & \theta^i & \psi^i \end{bmatrix}^T$ is the generalized coordinates of the body $i, i = 1,2,3$. There are three coordinate systems depicted, the global coordinate system, (X^0, Y^0, Z^0), represents a fixed reference frame, likely the ground in which the robot operates. The coordinate system labeled with superscript 1 $(\mathbf{x}^1, \mathbf{y}^1, \mathbf{z}^1)$ is fixed to the body of the robot (pendulum), this frame moves with the robot. The wheel coordinate systems, labeled with superscripts 2 and 3, $(\mathbf{x}^2, \mathbf{y}^2, \mathbf{z}^2)$ for the right wheel and $(\mathbf{x}^3, \mathbf{y}^3, \mathbf{z}^3)$ for the left wheel. These frames rotate with the wheels and are crucial for understanding how the wheels' rotation affects the robot's balance and movement.

*This work was not supported by any organization
[1]Benha Faculty of Engineering, Benha University, 13512 Benha, Egypt. ibrahim.abdulhadi@bhit.bu.edu.eg , mona.elawa@bhit.bu.edu.eg
[2] Egypt University of Informatics (EUI), New Administrative Capital, Cairo, Egypt. nader.mansour@eui.edu.eg
[3]Department of Robotics and Mechatronics, Egypt-Japan University of Science and Technology E-JUST, Alexandria 21934, Egypt. ayman.nada@ejust.edu.eg

979-8-3503-7397-4/24/$31.00 ©2024 IEEE