*applied sciences*

MDPI

*Article*

# Study on a Modular and Reconfigurable Robot That Can Be Used in Natural Disaster Events

**Andreea Dana Alionte, Lucian Bogatu, Cristian Gabriel Alionte *, Marius-Valentin Gheorghe, Vlad Andrei Stanescu and Robert Ciobanu**

Department of Mechatronics and Precision Mechanics, Faculty of Mechatronics and Mechanical Engineering, National University of Science and Technology Politehnica Bucharest, Splaiul Independenței nr. 313, Sector 6, RO-060042 Bucharest, Romania; ada.alionte@gmail.com (A.D.A.); lucian.bogatu@upb.ro (L.B.); marius.valentin03@gmail.com (M.-V.G.); vlad.stanescu1507@upb.ro (V.A.S.); robert.ciobanu@upb.ro (R.C.)
* Correspondence: cristian.alionte@upb.ro; Tel.: +40-726247296

**Abstract**

This paper presents a modular reconfigurable robotic system for search-and-rescue operations in natural catastrophe scenarios, addressing the structural rigidity, limited energy autonomy, and restricted victim-support capabilities of conventional rescue robots. The contributions are organized into four dimensions. First, a modular mechanical architecture is proposed in which modules operate independently or are interconnected through a multifunctional flexible tube providing mechanical coupling, shielded communication, wireless power transfer, and transport of vital supplies; a two-degree-of-freedom leg module enables star-, snake-, worm-, and chain-like configurations. Second, a unified locomotion model for both wheeled and legged configurations is developed through an equivalent-radius formulation, and stability is analyzed using the Zero-Moment Point (ZMP) criterion and the Linear Inverted Pendulum Model (LIPM). Third, a linear control law is formulated to compensate for longitudinal, lateral-slip, and orientation errors, while module coordination is achieved through a de-centralized, fault-tolerant status-message protocol. Fourth, three proof-of-concept prototype versions of modular reconfigurable robots with servo-driven active interconnections were manufactured by additive technologies. For Version 3, a Virtual Reality workflow integrating SolidWorks (version SOLIDWORKS 2023), Blender (version Blender 4.4.3, Unreal Engine 5 version 5.8), and Unreal Engine 5, with full-body inverse kinematics and physics-based simulation, was used to evaluate sensor-driven motion and obstacle avoidance. The results demonstrate the feasibility of a scalable platform able to adapt its locomotion and functional configuration to post-disaster environments

**Keywords:** modular robot; reconfigurable robot; search and rescue robotics

Academic Editors: Yutaka Ishibashi and Alessandro Di Nuovo

Received: 8 September 2026
Revised: 25 October 2026
Accepted: 30 October 2026
Published: 3 November 2026

**Copyright:** © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

## 1. Introduction

Robotic systems are progressively employed during natural disasters due to their primary advantage of safeguarding rescue personnel while navigating hazardous in-door environments [1,2]. Rescue robots are fundamentally utilized to execute some specific functions [3]. An essential task is recognition and must be executed in the impacted region to evaluate the circumstances for the effective deployment of search and rescue personnel [1]. The potential victim search feature is an essential responsibility because victims must

be identified initially to enable their rescue. Robots can traverse debris more effectively than humans, facilitating quicker victim recovery by rescuers. Rubble removal operations in search and rescue contexts markedly differ from those in construction or demolition endeavors. Rescue robots can assist rescuers in this endeavor. Telepresence, alongside psychological assistance, telemedicine can be highly advantageous. This pertains to a rescue robot that establishes a link between the rescue team operating outside the debris and the individual trapped beneath it [4].

Robots designed for natural catastrophe scenarios can be classified according to several criteria, including size, payload capacity, operating environment, locomotion type, degree of autonomy and human–robot interaction mode. From the perspective of post-disaster deployment, locomotion type is the most relevant criterion, since it determines the robot's ability to access victims through debris, narrow voids and uneven terrain; the following overview is therefore organized according to this criterion.

From the perspective of mechanical implementation, recent developments in additive manufacturing have expanded the possibilities for producing lightweight, compact, and geometrically complex components for robotic systems. Additive manufacturing is increasingly being investigated not only for rapid prototyping but also for the fabrication of functional robotic mechanisms that can subsequently be subjected to experimental characterization. Recent research has demonstrated additively manufactured compact transmission systems designed for robotic actuation, including eccentric planetary and planetary–cycloidal architectures. Functional prototypes have been experimentally characterized with respect to parameters such as reduction ratio, mass, dimensions, backlash, torsional behavior, friction, and back drivability [5,6]. These studies demonstrate the technological feasibility of using 3D-printed components in functional compact transmission mechanisms, although their suitability for a specific disaster-response robot must still be evaluated according to the mechanical loads, environmental conditions, reliability, and durability required by the application.

Existing rescue robots can be grouped according to their locomotion mode. Wheeled platforms are mechanically simple, fast and easy to control, and omnidirectional variants can maneuver in confined spaces; however, their mobility degrades significantly to rubble, steep slopes and discontinuous terrain [7,8]. Legged robots, ranging from bipeds and quadrupeds to hexapods and multi-legged designs, can step over obstacles, climb stairs and select their contact points, at the cost of higher mechanical and control complexity, greater energy consumption and higher cost [9]. Aerial robots provide rapid situational awareness, real-time imaging and access to areas unreachable from the ground, but are limited by flight endurance, airspace regulations and weather conditions, and cannot penetrate the confined voids inside collapsed structures [10–13]. Biomimetic robots inspired by snakes, worms or insects can move through narrow spaces and adapt to irregular surfaces but require complex control and are difficult to make robust for prolonged operation in harsh environments [14,15]. Each of these platforms is therefore efficient in a specific environment, whereas post-disaster sites combine open areas, debris fields, narrow voids and vertical obstacles within the same mission.

A great diversity of mobile robots that can be used in the event of natural disasters can be observed, with the main advantages and disadvantages being presented. The main analysis was carried out according to the mode of locomotion in the workspace, the focus being made on terrestrial robots. Particular attention is given to modular and reconfigurable architectures capable of employing different locomotion mechanisms, since such adaptability may be advantageous when operating in the heterogeneous and unpredictable environments encountered during rescue missions.

Modular self-reconfigurable robots address this heterogeneity by assembling a structure from multiple modules that can change their topology to match the task and the terrain [16–18]. Representative chain-type and hybrid systems include PolyBot [19], CONRO [20], M-TRAN [21], SuperBot [22] and SMORES [23], which demonstrated snake-like, legged, rolling and wheeled gaits obtained from the same set of modules, as well as distributed control strategies in which modules coordinate through local communication [20]. In parallel, several serially connected robots have been developed specifically for search and rescue, such as the snake-like crawler KOHGA [24], the SOURYU series of connected crawler vehicles [25], the modular snake robots deployed after the 2017 Mexico City earthquake [26,27] and the Active Scope Camera, a flexible cable-like robot inserted into rubble [28].

The comparison highlights several gaps. (i) General-purpose modular self-reconfigurable robots have mainly been studied as laboratory platforms for reconfiguration and gait-generation algorithms, and their use in real rescue scenarios remains limited [29,30]. (ii) Rescue-oriented serial robots are usually built with a fixed topology and a single locomotion mode, so their configuration cannot be adapted to the access route once deployed. (iii) In both classes, energy autonomy is limited by onboard batteries, and where a tether is used, it serves only as a passive power or data cable, not as a structural element that can guide modules, transmit energy contactless and convey supplies. (iv) Existing systems are designed almost exclusively for exploration and victim localization, with no means of establishing a durable communication and material-support link with trapped persons [31,32]. (v) Kinematic and stability models are generally formulated separately for wheeled and legged modules, which complicates the control of heterogeneous configurations combining both loco-motion types.

To address these gaps, this paper proposes a modular reconfigurable robot in which the modules are linked by a multifunctional flexible tube that acts simultaneously as a mechanical connection, a shielded communication channel, a contactless power-transfer path and a conduit for vital supplies. The leg module allows passive and active reconfiguration into star-, snake-, worm- and chain-like structures. A unified kinematic model based on an equivalent-radius parameter is developed for both wheeled and legged modules, together with ZMP- and LIPM-based stability analysis (gap v). Finally, the system is implemented as a 3D-printed prototype and a VR simulation environment, as a step toward deployment in realistic rescue scenarios.

It should be emphasized that the system presented in this paper is a laboratory proof-of-concept prototype, intended to validate the modular architecture, the unified kinematic model, the control strategies and the reconfiguration principle, rather than a field-ready rescue platform. The targeted application scenario involves inspecting, delivering material, and supporting communication inside buildings that have collapsed after earthquakes, using openings that are comparable in size to the cross-section of a module. The main characteristics of the current prototype are summarized in Table 1. Several gaps remain before deployment at real disaster sites: (i) the 3D-printed polymer structure and the commercial servomotors have not been qualified for the loads, impacts, dust, moisture and temperature ranges encountered in rubble; in particular, the MG996R (Tower Pro Pte Ltd, T 5 Bedok Reservoir View, Aquarius by the Park, #14-0, Singapore, Singapore 478928) servomotors are not rated for continuous operation close to their stall torque, and the legs and connection elements may snag on loose rubble; (ii) obstacle-surmounting capability, load-bearing capacity and endurance have so far been assessed only on laboratory surfaces; (iii) the contactless power-transfer and shielded-communication functions of the tube are presented at concept level and have not yet been experimentally characterized; (iv) localization, perception in dark and dusty environments and long-range teleoperation

inside rubble have not been addressed; and parameters of the trajectory-tracking law remains to be calibrated experimentally. These aspects define the directions of future work.

**Table 1.** Main characteristics of the laboratory prototype (Version 3).

| Parameter | Value |
|---|---|
| Degrees of freedom per leg module | 2 |
| Joint range | ±90° |
| Actuators | MG996R digital servomotors |
| Actuators stall torque | 11 kgf·cm (@ 6 V) |
| Maximum number of interconnected leg modules (same type) | 4 |
| Controller | Maker Pi RP2040 (Cytron Technologies Sdn Bhd (Penang HQ), Adress: Lorong Industri Impian 1, Taman Industri Impian, 14000 Bukit Mertajam, Pulau Pinang, Malaysia )(RP2040 microcontroller) |
| Power supply | 1-cell Li-ion/LiPo battery (Gens ace 200mAh 3.7V 45C 1S1P, Manufacturer: Shenzhen Grepow Battery Co., Ltd., Adress: Huarong Road, Gaofeng Community, Dalang Street, Longhua District, Guangdong, China ) or wired supply |
| Structural material | PLA/wood |
| Mass of one leg module only type A | 155 g |
| Mass of one leg module type A and B | 255 g |
| Total mass (central body + legs) only modules type A | 1200 g |
| Total mass (central body + legs) modules type A and B | 1850 g |
| Overall dimensions (L × W × H) only modules type A | 210 × 150 × 400 mm |
| Overall dimensions (L × W × H) modules type A and B | 210 × 150 × 440 mm |

## 2. Materials and Methods

The modular reconfigurable robot is an adaptable device designed to assist individuals trapped in fallen buildings during natural disasters, such as earthquakes. Typically, the robots used in emergency situations consist of a single unit that functions independently and communicates wirelessly with rescue teams. However, existing solutions have several limitations: they feature a rigid design that is not adaptable to various terrains, are intended solely for exploration, cannot provide assistance to endangered individuals, and have limited energy capacity.

A key technical concern is the lack of material support and communication resources for potential victims in areas that are unreachable due to structural collapse following a natural disaster. Additionally, there is an inadequate adaptation of robotic designs to the differing conditions encountered on access routes to potential victims in inaccessible sections of destroyed buildings, along with the necessary support tasks required for these individuals.

We present a solution that overcomes the drawbacks of current systems by enabling the establishment of a material support and communication network for potential victims located in inaccessible areas of collapsed structures. It also offers an adaptable robotic

framework capable of providing various configurations tailored to on-site requirements. An important advantage of implementing the solution is that the system assists in conveying materials relevant to these scenarios, whether internally within the modules or externally attached. This aids communication with potential victims in hard-to-reach areas within fallen edifices, delivers modular robotic systems that can be customized to meet specific needs, ensures the system is always charged during operation, and provides a versatile robotic module that facilitates the implementation of various mechanisms for movement, item handling, or transportation.

### 2.1. Structure of a Robotic Module

The structure of a robotic module, as shown in Figure 1, features a central body that consists of a frame to which various surfaces can be attached; these surfaces may include fixed systems for locomotion, object handling, sensors, and electronic systems for the command and control of moving elements located on the side surfaces. Locomotion systems can use different elements for movement that can be diverse (wheels or legs), and in this article we will focus on locomotion systems consisting of one independent reconfigurable module with legs of movement that have two degrees of freedom.

The modular reconfigurable robotic system is designed for use in prospective earthquake scenarios, but the modules can be combined to make a multimodule reconfigurable robotic system using a flexible tube. Several robotic modules can be connected using a flexible tube, and therefore a module can connect, disconnect from the tube, and pull and push the tube. The robotic module initially places the tube, which serves as the foundation for the multimodule robotic system designed for a prospective earthquake. The tube is initially placed by operating the robotic module. The tube fulfills various roles. The main objective is to create a communication pathway with the person in turmoil. The tube will segregate and eradicate electrical, magnetic, or other interferences, functioning as a safeguarded communication channel, thus facilitating the conveyance of messages (audio, video, or data) over radio or light waves while reducing the effects of disturbances. Another role is to enable the conveyance of more robotic modules, which could be crucial for supplying diverse materials necessary for the person's existence or to improve their comfort. Another role is to continuously power robotic modules using contactless technologies, such as wireless energy transfer. The reconfigurable robotic framework consists of various modules that can be re-arranged and linked through a multifunctional tube: transporting vital survival items, enabling communication with interference shielding, and delivering power to robotic components. This multimodule reconfigurable robotic system will be presented in several future work.

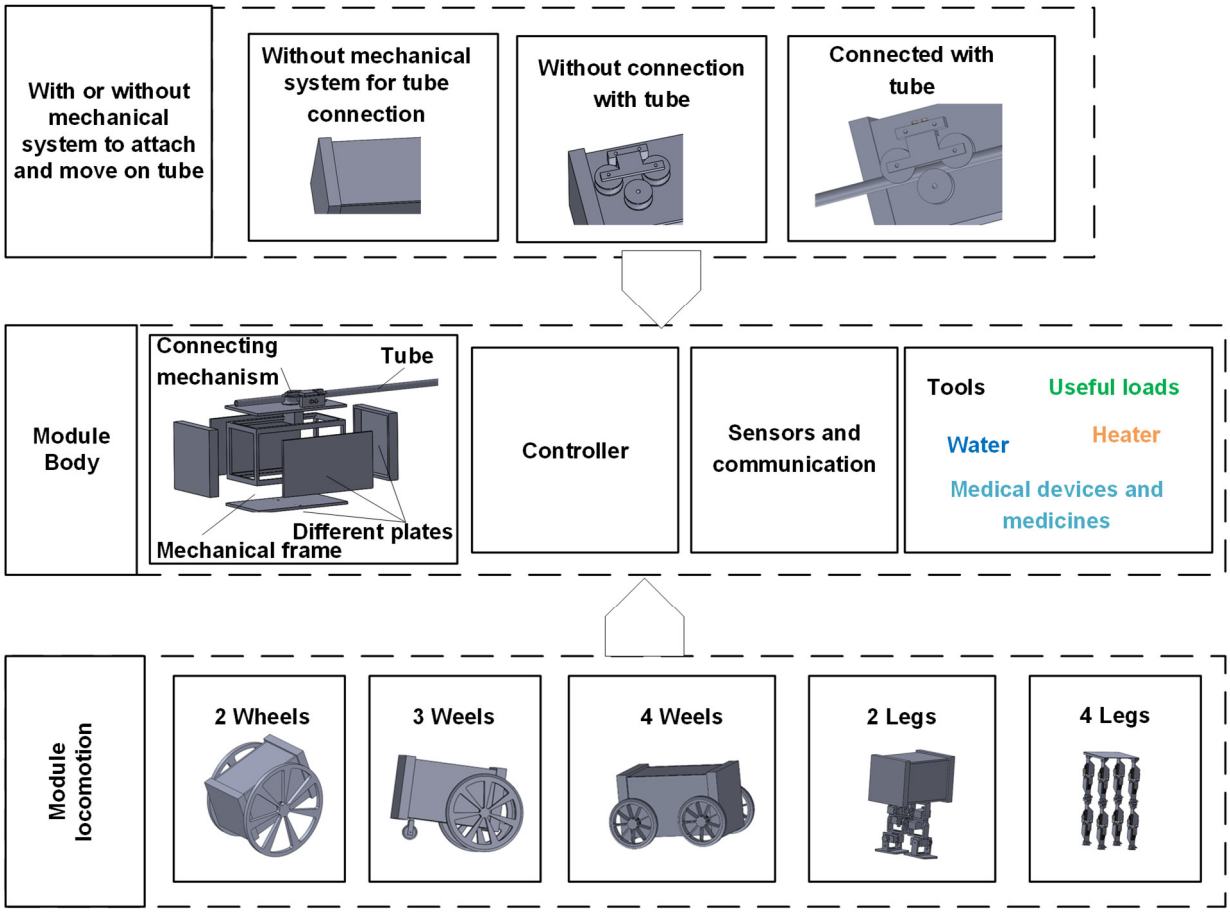



**Figure 1.** Modular reconfigurable robot structure.

The proposed structure offers considerable adaptability. At both the singular robotic module tier and the composite level consisting of many modules. The locomotion mechanism for each scenario will be analyzed in greater detail, focusing on displacement regardless of the structure. This will involve a stationary tube with a tail, as well as a configuration featuring two robotic modules interconnected by a tube.

### *2.2. Independent Locomotion of the Robotic Module*

Displacement must be studied because it is a fundamental skill required in a wide range of applications. By displacement, we mean the action of positioning from one place in space to another. Self-configuring robots have the advantage that depending on the environment, they can change their shape to move optimally and can consequently change their displacement pattern. The locomotion of a mobile robot represents its ability to move from one place to another in space and is directly influenced by the movement of the mechanism that generates the movement. Locomotion depends on the environment in which it takes place, and robots are named after it. Classification of locomotion can be based on several criteria: the type of movement of the final effector of the locomotion mechanism (such as rotation, rolling, or translation in horizontal, vertical, or angled directions); the number of kinematic couplings (2, 3, 4, etc.); the number of active motor couplings (2, 3, 4, etc.); and the distance of the active element from both the basic structure of the robot (the chassis) and the final element (such as a foot or wheel), which can be located on the chassis, at a chassis element, or at the level of the element in contact with the ground.

The locomotion can be performed either through the contact between the soil and the robot's body (which is bio-inspired or biomimetic: snake or earthworm), by intermittent

contact between the ground and the feet (which are bio-inspired or biomimetic: spider, salamander, kangaroo, human, etc.), or by the permanent or quasi-permanent contact between the ground and the wheels (which represents the easiest, fastest, and most efficient locomotion but does not exist in nature). We can consider the independent module as a mobile robot, and thus we can consider the following types of locomotion based on the elements involved in movement (wheels or legs): with two elements, with three elements, with four elements, with six elements, or with more than six elements. In this article, we will detail movement with four elements. The efficiency and ease of locomotion are determined by the number of active couplings, the complexity of the structure, the complexity of the control, and energy efficiency (it also depends on the terrain shape). Therefore, in designing a robotic module, we considered the following design phases. The first phase is to determine the mechanical constraints, and there are numerous options when it comes to designing a mobile robot, with the main factor in determining the construction variant being dimensions, degrees of freedom, joints, and stability. The second phase is to determine the dimensions that play a major role because of the project's cost, the materials needed for manufacturing, and the drive systems are determined based on the sizing. The third phase is to identify the degrees of freedom at the robotic module level that influence the mechanical structure and the implementation of robot control for different locomotion environments. Degrees of freedom directly influence the number and type of joints.

The next step is to analyze stability. In the desire to reach a point where the robot, whether at rest (static) or in motion (dynamic), is in a position of balance or stability, the locomotion elements will be constructed and controlled to generate movement in certain sequences under conditions in which the robot will need to maintain its balance. If the robot has too much sway that does not allow for the stable maintenance of its center of gravity, additional weights will be added, or an inverted pendulum control model will be applied. The next step is to analyze the ground contact surface. The stability of the walking robot is determined by the ground of its foot or the contact point with the element (wheel, track, etc.) that encounters the running surface. In general, the design trend is towards oversized and heavy soles/wheels to maintain balance due to a large contact surface. But this option has a drawback: the motor torque needed to move the parts will be very high.

The mode of movement can be independently determined before sending the robot on a mission by simply choosing the side walls that will have the locomotion systems attached. In the design of mobile robots, various kinematic and dynamic problems specific to the chosen structure arise. Kinematics is closely related to the structure of robots, as their construction has often proven to be an obstacle in developing new optimal solutions, where the material strength must allow for weight support and not introduce significant inertia into the system. Therefore, lightweight structures that are sufficiently strong are preferred so that the robot does not experience structural issues or control problems due to the flexibility of the legs. Sometimes, high-precision motors with high torque are necessary so that the structure does not require complicated gear mechanisms to increase torque and control complexity. The kinematics of mobile robots describes the relationship between how the actuating elements are moved and how the robot moves, considering some simplifying assumptions. We will assume that the robot module is made up of a single rigid body (although we can extend the model to a tube or other modules), the robot's location corresponds to the position and orientation of a coordinate system attached to its body, the robot instantly reaches the commanded speed (there is no dynamics/inertia), and the friction is ignored, and therefore, in each direction, the wheels either roll perfectly, and the contact between the feet and the ground is fixed.

When analyzing the kinematics of the robot, that is, how the robot's movement is defined by the parameters of displacement and rotation at the center of mass, namely

$\dot{q}(\dot{x}, \dot{y}, \dot{\theta})$ as a function of the rotation speed of the drive elements, the problem can be divided into several parts:

- Identifying the relationship between the robot's movement in its body-attached coordinate system and the coordinate systems of the elements that create the motor's displacement is crucial. These coordinate systems are usually established at the centers of rotation.
- The relationship between the movement of the wheels and the movement of the robot's body is identified. Typically, a matrix representation is obtained by starting from the coordinate systems of the mobile elements and progressing to the robot's coordinate system.
- The relationships through which the body translates or rotates are written based on the rotation or translation of the individual drive elements, and the constraints of the individual drive elements (wheels or legs) are identified.
- The relationships that define inverse kinematics are determined, meaning that when the robot body moves, the parameters of the drive elements (wheels or legs) change accordingly, and this process explains how those parameters are affected.
- The inverse of the inverse kinematic equation can be used to verify direct kinematics.

A term closely related to robot locomotion is mobility, because mobile robots are generally those that can move from one place to another. Mobility offers a robot much greater flexibility to perform new, complex, and important tasks. Highly mobile robots can perform more natural tasks in environments that are not specially designed for them. These robots can work in a human-centered space and can cooperate with users by sharing a workspace together. A mobile robot needs locomotion mechanisms that allow it to move freely in its environment. There is a wide variety of possible modes of movement, which makes the choice of a robot's locomotion approach an important aspect of mobile robot design.

Since controlling a mobile robot requires a reference in the operational space where the robot moves, it is necessary for these data to be transformed from the Cartesian coordinate system into values in the joint space through inverse kinematics calculation methods. For this, geometric methods for solving inverse kinematics can be used, as well as the method based on the transposed Jacobian matrix, to avoid the problem of singularity points due to the inversion of the Jacobian matrix.

The use of geometric methods is possible for kinematic chains utilized with up to three degrees of freedom. It results in the possibility of calculating three unknown values from three trigonometric relationships, provided certain conditions are imposed for choosing multiple solutions and avoiding points of division by zero. One of the recent solutions for solving inverse kinematics is the calculation algorithm that uses the transposed Jacobian matrix presented in mathematical relations.

The major advantage of this solution is the elimination of the problem of singularity points, whose effect can lead the system toward states of instability. The disadvantage of this method is the approximate calculation of joint errors. In contrast, the calculated error has a linear character, making it successfully usable with linear control laws. This algorithm, which uses the transposed Jacobi matrix, is extremely useful when the control system is linear, as it significantly reduces the computational power required to solve the inverse system and does not need additional conditions for choosing optimal solutions or eliminating singularity point issues. Instead, it cannot be used as a reference for nonlinear control laws, as they require the exact value of the reference in joint space. In addition to the advantages and disadvantages of the inverse kinematics calculation algorithm using the transposed Jacobian matrix, the accuracy of the positioning error of the controlled robot joints must be calculated, which can be validated through experiments.

### *2.3. Locomotion Kinematics of a Robotic Module*

In order to analyze the kinematics of the robot module, we will consider the notations in Figure 2 in which the module has a rigid body, the displacement elements maintain the same vertical position of the robot (the vertical displacement of the pegs maintains a vertical position or the wheels are non-deformable). Also, we assume that the robotic module moves on a plane without slipping having the generalized coordinates ($x$, $y$, $\theta$) of the center O of it, where $x$ represents the position on the $X_G$-axis, y represents the position on the $Y_G$-axis, and $\theta$ represents the orientation angle, with respect to the global coordinate system ($O_G$, $X_G$, $Y_G$).

The robotic module moves with the velocity $v$ oriented in the X-direction. Thus, we have the coordinate system attached to the robotic module ($O$, $X$, $Y$), and the parameters: the length between axes is denoted with $L$, $v_d$ is the speed of the right displacement element, $v_s$ is the speed of the left displacement element, and $\omega$ is the angular velocity of the robot around the center $O$. Of course, it is important to introduce a new constant related either to the radius of the wheels or to the distance between the axis of rotation of the perpendicular to the contact surface between the foot and the ground and the point of intersection between this perpendicular and the rotational surface. It is like replacing the entire leg of the robot with an equivalent wheel. This assumption ignores stance/swing phase dynamics, variable ground clearance, leg lift-off, and impact forces.

The direct kinematic model of the robot module is based on state parameters [33]:

$$\begin{aligned} \dot{x} &= \frac{1}{2}(v_d + v_s)\cos(\theta) \\ \dot{y} &= \frac{1}{2}(v_d + v_s)\sin(\theta) \\ \dot{\theta} &= \frac{1}{L}(v_d - v_s) \end{aligned} \tag{1}$$

Alternatively, the robot kinematics equations described above can be expressed in terms of $v$ and $\omega$ as:

$$\begin{aligned} \dot{x} &= v\cos(\theta) \\ \dot{y} &= v\sin(\theta) \\ \dot{\theta} &= \omega \end{aligned} \tag{2}$$

The wheels determine how the robot interacts with the ground. Each wheel type has one or more control parameters and one or more constraints depending on the control type and wheel type. The constraints depend on the construction of the wheel and the assumptions made (such as the fact that there is no slippage). The wheel model describes, mathematically, the motion of the wheel and is parameterized by the variables of the wheel (such as angular velocity). By replacing the wheel model in the wheel-body-motion equations and solving the wheel variables, we determine which control parameter values are required to achieve a certain motion and which displacements are achievable and which are impossible to achieve.

https://doi.org/10.3390/app162110915

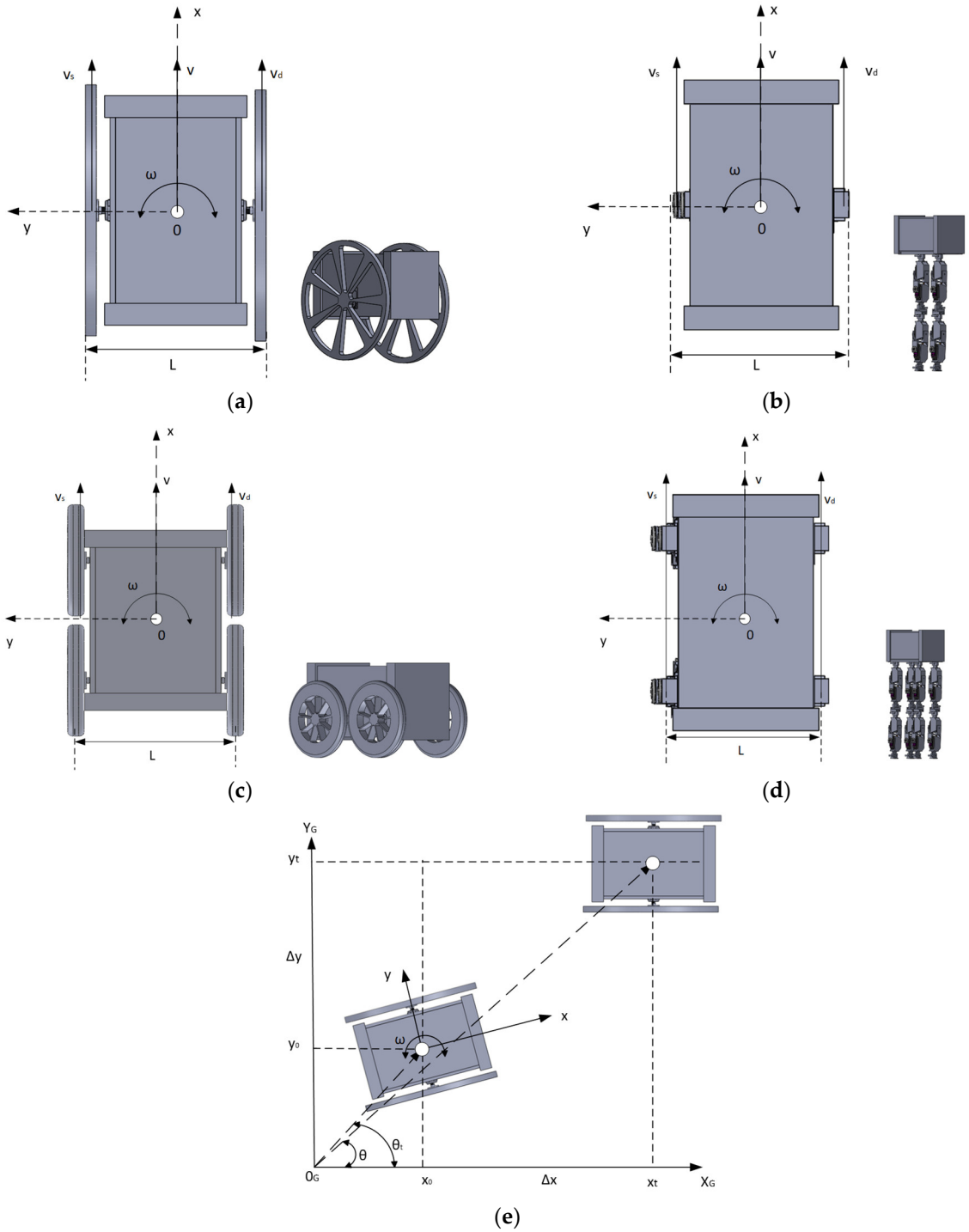



**Figure 2.** Robotic module: (**a**) top view of a two-wheeled robotic module; (**b**) top view of a two-legged robotic module, (**c**) top view of a four-wheeled robotic module; (**d**) top view of a four-legged robotic module, (**e**) locomotion of a robotic module with two displacement elements with respect to a reference coordinate system.

Stepping robots have the following characteristics: they can also move on rough terrain; it requires low energy consumption; the high ground clearance allows them to overcome obstacles; the contact with the ground is discontinuous; the foot has the possibility to select the point of contact (support); they have the possibility of perceiving contact with the ground; they can move on soft ground; they damage the soil on which they move a little (an important aspect in forest exploitation); the travel speed is low; and the driving control is complicated due to the high number of degrees of freedom. If we consider the

two-legged module, we have three phases. One phase is a swing in which the robot changes its center of gravity to the side on one leg to allow the opposite leg to detach from the ground. Some robots can stay on one leg for a longer time, the leg used for moving having a longer stroke, having a practically unlimited time to move when it is in the air, and others stay for a limited time on one leg, the leg used to move having a reduced and fast stroke while in the air.

The second phase involves positioning the foot in the air, during which the point of contact between the foot and the ground will rotate around an axis parallel to that of the body, similar to the angular velocity ω; this rotation can result either from the movement of a part of the foot or from the combined rotations of the elements of the leg. In the third phase, the actual step will take place. Taking into account only the beginning and end of the step cycle, we can consider that the displacement is similar to that of wheeled modules. In most cases, the left wheels of the four-wheeled robot will rotate simultaneously, as will the right wheels. This allows the model from the two-wheeled robot to be applied.

In the direct kinematic analysis of the stepping robot, there are some simplifying assumptions: the leg robot consists of $n$ rigid components; all kinematic information, such as the position of the center of gravity, the orientation of the parts, the speeds, etc., is known and calculated by the direct kinematic method; the ground is rigid and static; the leg does not slip on the ground; and all joints are actively operated. In the case of robots with legs, based on the above conditions, we can equate legs with equivalent wheels. Thus, regardless of the number of wheels or legs, we can explain the speed of each element of motion that will rotate on the robot's plane of motion, and we will write the equation that moves with the linear velocity $v$ and rotates with the angular velocity $\omega$. Leg locomotion has the advantage that it offers the possibility of overcoming obstacles, but it is energy-inefficient and requires a good control. Only the points of contact matter; the terrain between contacts has no influence.

We will obtain the equations of the two wheels whose command values are defined by the speeds $v_d$ and $v_s$

$$\begin{aligned} v_d &= \frac{2v + \omega L}{2} \\ v_s &= \frac{2v - \omega L}{2} \end{aligned} \tag{3}$$

For the robot to reach a point in space, the matrix form of the equation of state is considered

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{pmatrix} = \begin{pmatrix} \cos(\theta) & 0 \\ \sin(\theta) & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} v \\ \omega \end{pmatrix} \tag{4}$$

We consider that the robot is moving towards the target position defined by the coordinates $x_t$, $y_t$ and $\theta_t$. The robot is commanded to move toward the target position by continuously correcting the position and orientation errors, which, in the global coordinate system, are:

$$\begin{aligned} \Delta x &= x_t - x \\ \Delta y &= y_t - y \\ \Delta\theta &= \arctan\big(sin(\theta_t - \theta), cos(\theta_t - \theta)\big) \end{aligned} \tag{5}$$

However, to properly control the motion of a differential-drive robot, it is useful to express the position errors in the robot's coordinate system:

$$\begin{gathered} dx = \Delta x \cdot \cos(\theta) + \Delta y \cdot \sin(\theta) \\ dy = -\Delta x \cdot \sin(\theta) + \Delta y \cdot \cos(\theta) \\ d\theta = \Delta\theta \end{gathered} \tag{6}$$

The distance to the destination is calculated using the equation:

$$\rho = \sqrt{(\Delta x)^2 + (\Delta y)^2} \tag{7}$$

The direction toward the destination and the angular error relative to this direction are calculated using the following equations:

$$\begin{gathered} \theta_d = \arctan(\Delta y, \Delta x) \\ \alpha = \arctan\left(sin(\theta_d - \theta), cos(\theta_d - \theta)\right) \end{gathered} \tag{8}$$

The following robot control laws can be implemented using two proportional controllers:

$$\begin{gathered} v = K_\rho \rho \\ \omega = K_\alpha \alpha \end{gathered} \tag{9}$$

where $K_\rho > 0$ and $K_\alpha > 0$ are tuning gains.

After determining the robot's linear and angular velocities, $v$ and $\omega$, the linear velocities of the wheels are calculated using the kinematic relationships:

$$\begin{gathered} v_d = K_\rho \rho + \frac{LK_\propto \propto}{2} \\ v_s = K_\rho \rho - \frac{LK_\propto \propto}{2} \end{gathered} \tag{10}$$

Once the robot has reached the vicinity of the target point with coordinates ($x_t$, $y_t$), within a specified position error $\varepsilon_p$, that is,

$$\rho \leq \varepsilon_p \tag{11}$$

The robot rotates in place with angular velocity:

$$\omega = K_f \Delta\theta \tag{12}$$

where $K_f > 0$ is the tuning gain of the final orientation controller.

Therefore, the linear wheel velocities are:

$$\begin{gathered} v_d = \frac{LK_f \Delta\theta}{2} \\ v_s = -\frac{LK_f \Delta\theta}{2} \end{gathered} \tag{13}$$

The rotation stops when the final orientation error is smaller than a specified tolerance $\varepsilon_\theta$, that is,

$$|\Delta\theta| \leq \varepsilon_\theta \tag{14}$$

The kinematic model is subject to certain constraints due to its assumption that the terrain is stationary and rigid, which implies that it does not deform or move under the robot. As a result, it is unable to accurately simulate locomotion on soft or deformable surfaces, such as sand, snow, soil, or dirt. The predicted and actual robot motion may

result in discrepancies due to the incomplete capture of terrain deformation and the resultant differences in contact conditions. The no-slip constraint is another critical assumption, which stipulates that the velocity of a foot in contact with the ground is zero in relation to the earth, provided that the contact point remains stationary. This assumption does not consider traction loss on wet, icy, or loose surfaces and does not account for sliding or slipping during locomotion. The model may predict a stable gait even when the actual robot loses traction, and the estimated position and velocity of the robot may become inaccurate when slipping occurs. Additionally, the model makes the assumption that all joints are actively actuated, with each modelled joint being endowed with an actuator that can regulate its motion. Passive joints, springs, and compliant mechanisms are not explicitly accounted for by this assumption. Furthermore, actuator limitations, such as torque, speed, power, and bandwidth constraints, may be disregarded. Consequently, the model may forecast joint trajectories that are incapable of being physically realized, while energy consumption, actuator saturation, and motor dynamics are not accounted for. Under predictable contact conditions and relatively sluggish locomotion, the model is appropriate for kinematic analysis and gait planning, despite these limitations. The model can be used when the robot is operating on firm, stationary topography, the feet maintain sufficient traction, and the commanded joint motions remain within the robot's mechanical and actuation limits.

### *2.4. Stability of a Robotic Module*

The balance of a mobile robot is essential for ensuring stable and reliable operation during locomotion. Based on their stability characteristics, mobile robots can be classified into three main categories. Statically stable robots remain in equilibrium throughout their movement and always maintain at least two elements in contact with the ground. Quasi-statically stable robots may temporarily adopt an unstable configuration, but only for a very short period during locomotion before returning to a stable state. Dynamically stable robots, in contrast, do not maintain a stable configuration throughout their movement and rely on dynamic effects to maintain balance during locomotion.

The static balance of a mobile robot can be checked with the help of the support polygon in the case of a robot with four or more travel elements, which is the polygon formed by the horizontal projections of the support points of the wheels and legs during the support phase. In the case of two-wheeled robots, we have a support axis consisting of the horizontal projections of the support points of the wheels, respectively, of the legs in the support phase. The gait is statically stable if, at any time, the vertical projection of the center of gravity G is inside the support polygon or on the support axis. This applies to both the front and side planes. A problem that can only occur in stepping robots is that of geometric interference of the legs. If the leg travel is greater than the distance between two adjacent legs, the leg workspaces intersect, which means that interference is possible. Geometric interference of the legs can be avoided by properly correlating some parameters that characterize the gait.

From a mathematical point of view, the above condition starts using the criterion of projecting the point corresponding to the center of gravity, which assumes that a body that is not in motion is confronted only with the gravitational force, which is exerted in all the components of the robot. These forces can be replaced by a virtual force, denoted with *R*, acting as the equivalent of the robot's center of gravity (denoted with *CoM*) at the $p_{COM}$ point that can be found by:

$$p_{COM} = \frac{\sum_{i=1}^{n} m_i p_i}{\sum_{i=1}^{n} m_i} \quad (15)$$

The robot is considered to have $n$ components, each having the mass $m_i$ and the vectors $p_i$, which are the individual distances of the centers of gravity for each element. The robot is in equilibrium if, considering the gravitational acceleration $g$, the relation:

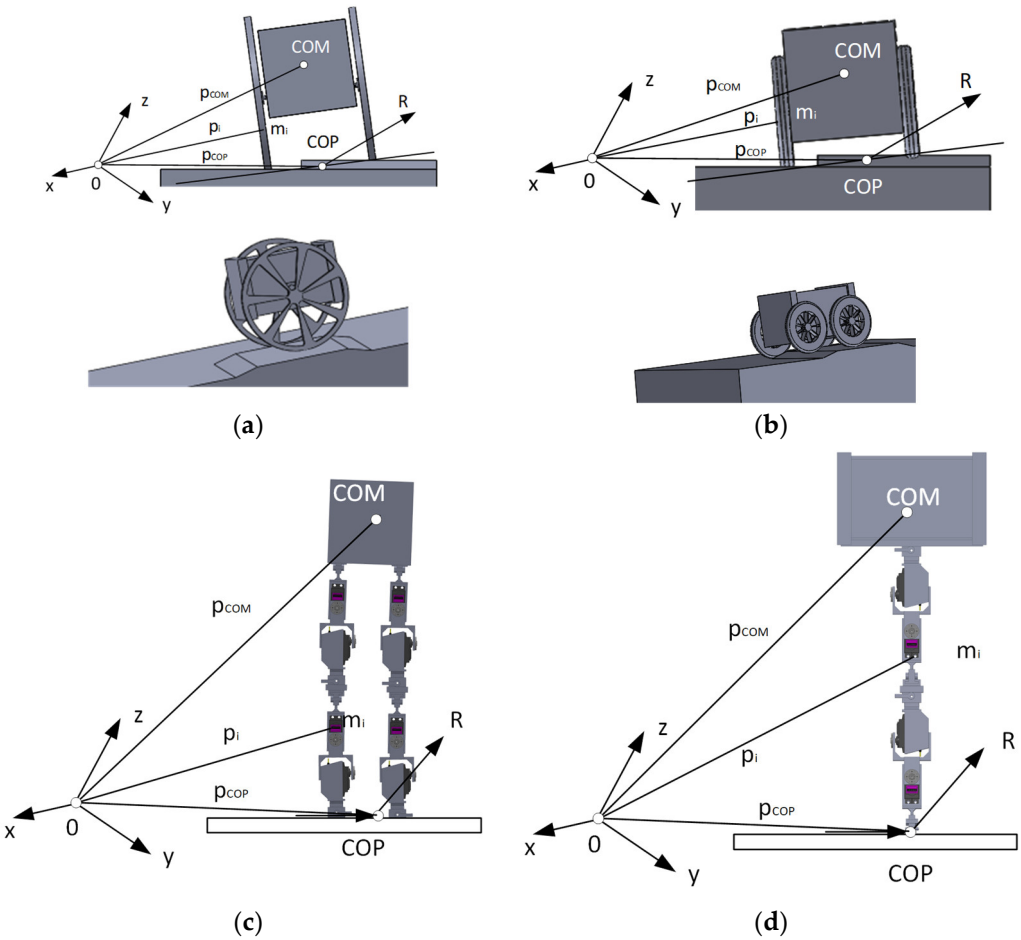
$$\sum_{i=1}^{n}[(p_{COM} - p_i) \text{ x } m_i g] = 0 \tag{16}$$

If the ground projection of $p_{COM}$ remains in the support polygon or on the support axis, the moving robot will not fall. However, when motion becomes rapid, dynamic forces will dominate static forces, and this criterion is not sufficient [33].

Another criterion is that of the center of pressure (denoted *CoP*), which is the point in the support polygon where the total sum of the contact forces, *R*, acts, producing a force but not a moment. In the static equilibrium position, the components of the robot that are used and considered are the soles. Under the effect of the contact forces, two forces will be generated, one of friction, which is zero because the fulcrum does not slide on the surface of the ground, and the pressure force in the normal direction (denoted with $F_N$). The resultant on the normal direction of the pressure field with $n$ points of contact denoted with $p_{FNi}$ and the position of the point CoP (denoted with $p_{COP}$) are:

$$\begin{aligned} F_{RN} &= \textstyle\sum_{i=1}^{n} F_{Ni} = 0 \\ p_{COP} &= \frac{\sum_{i=1}^{n} p_{FNi} \cdot F_{Ni}}{R} \end{aligned} \tag{17}$$

If the center of pressure is outside the support polygon, the robot will fall off. Consider the distance from the origin of the motion system to the center of gravity, $p_{COM}$, according to the above relationship and the graphical representation in Figure 3.




https://doi.org/10.3390/app162110915

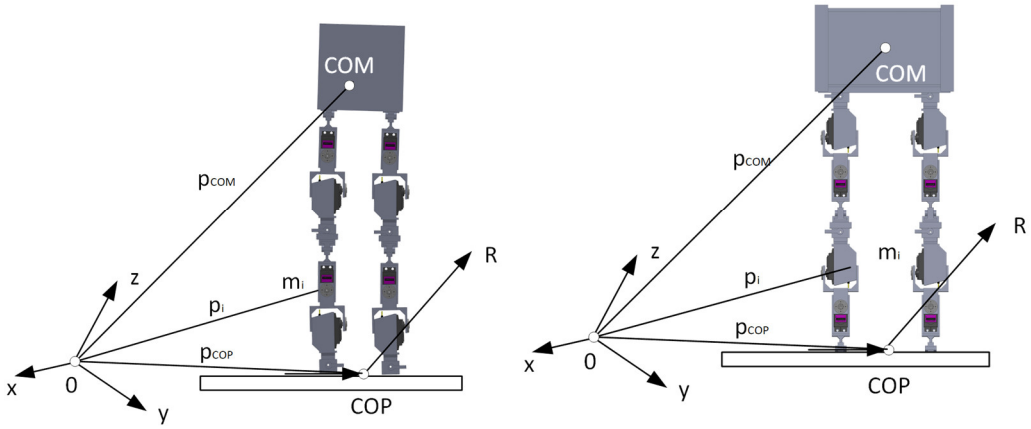



**Figure 3.** Stability of a robotic module: (**a**) front view of a two-wheeled robotic module; (**b**) front view of a four-wheeled robotic module, (**c**) front view of a two-legged robotic module; (**d**) side view of a two-legged robotic module, (**e**) front view of a four-legged robotic module; (**f**) side view of a four-legged robotic module.

We define the total inertial momentum, $T$, and the total angular momentum, $H$, with reference to the $OXYZ$ system, considering the inertia tensor, $I_i$, and the angular velocity, $\omega_i$, of component $i$ with respect to the reference system and calculate the derivative:

$$\begin{cases} T = \sum_{i=1}^{n} m_i \dot{p}_i \\ H = \sum_{i=1}^{n} p_i \times m_i \dot{p}_i \times I_i \omega_i \end{cases} \Rightarrow \begin{cases} \dot{T} = \sum_{i=1}^{n} m_i \ddot{p}_i \\ \dot{H} = \sum_{i=1}^{n} [\, \dot{p}_i \times m_i \dot{p}_i \times I_i \omega_i + p_i \times m_i \ddot{p}_i \times I_i \omega_i + p_i \times m_i \dot{p}_i \times I_i \dot{\omega}_i] \end{cases} \quad (18)$$

where $\dot{p}_i \times m_i \dot{p}_i$ equals zero.

$$\begin{aligned} R = -S = \dot{T} - mg \\ M = \dot{H} - p \times mg = p_{OP} \times R + M_{COP} \end{aligned} \quad (19)$$

$S$ and $M$ represent the force and moment that describe how the surface on which the robot walks reacts to it. The total sum of the contact forces, $R$, acts at the point $CoP$, $M_{COP}$ is the moment at the point $CoP$, and $p_{COP}$ is the vector from the origin of the reference system to the point $CoP$. Because $M_{COP} = [0, 0, M_z]$ is at point $CoP$, being ZMP (zero-moment point) or FZMP (zero-moment point with friction), we have:

$$M_p = \dot{H} - p_{COM} \times mg + R \times p_{COP} \quad (20)$$

Starting from the equation, we can find the distance from the origin of the reference system to the zero-moment point or the zero-moment point with friction, $p_{ZMP} = p_{COP} = [x_{ZMP}, y_{ZMP}, z_{ZMP}]^T$, and we have:

$$\begin{aligned} x_{ZMP} = \frac{m \cdot g_z \cdot p_{COMx} - z_{ZMP} \cdot \dot{T}_x - \dot{H}_y}{m \cdot g_z + \dot{T}_z} \\ x_{ZMP} = \frac{m \cdot g_z \cdot p_{COMy} - z_{ZMP} \cdot \dot{T}_y - \dot{H}_x}{m \cdot g_z + \dot{T}_z} \end{aligned} \quad (21)$$

where $z_{ZMP}$ is the height of the ground relative to the ground. But the $OXY$ plane is placed on the ground, so $z_{ZMP}$ is null. The following Huang equations are obtained [30]:

$$x_{ZMP} = \frac{\sum_{i=1}^{n}\{m_i[p_{ix}\,(\ddot{p}_{iz} + g_z) - p_{iz}\,(\ddot{p}_{ix} + g_x)] - I_{iy}\omega_{iy}\}}{\sum_{i=1}^{n} m_i\,(\ddot{p}_{iz} + g_z)}$$
$$y_{ZMP} = \frac{\sum_{i=1}^{n}\{m_i[p_{iy}\,(\ddot{p}_{iz} + g_z) - p_{iz}\,(\ddot{p}_{iy} + g_y)] - I_{ix}\omega_{ix}\}}{\sum_{i=1}^{n} m_i\,(\ddot{p}_{iz} + g_z)} \quad (22)$$

To ensure the stability of the robot, a simple controller can be implemented based on a $p_{COM}$ position, speed, and acceleration estimation system (which can be obtained experimentally using an IMU-type sensor located in the robot's central body), along with the inverse kinematic system and the coupling drive systems. The mathematical model is LIPM [34] and is based on the equations $d_{(COM,P),x}$ and $d_{(COM,P),y}$ which represent the projections on the x and y axes of the distance defined by the point CoP and CoM:

$$\ddot{d}_{(COM,P),x} = \omega^2(d_{(COM,COP),x} - p_{COM,x})$$
$$\ddot{d}_{(COM,P),y} = \omega^2(d_{(COM,COP),y} - p_{COM,y}) \quad (23)$$

where $\omega$ represents the natural frequency of the pendulum and is equal to:

$$\omega = \sqrt{\frac{g}{|d_{COM,COP}|}} \quad (24)$$

The equations can be transformed into the equation of state:

$$\begin{bmatrix}\dot{d}_{(COM,COP),x}\\ \ddot{d}_{(COM,COP),x}\end{bmatrix} = \begin{bmatrix}0 & 1\\ \omega^2 & 0\end{bmatrix}\begin{bmatrix}d_{(COM,COP),x}\\ \dot{d}_{(COM,COP),x}\end{bmatrix} + \begin{bmatrix}0\\ -\omega^2\end{bmatrix}p_{COM,x})$$
$$\begin{bmatrix}\dot{d}_{(COM,COP),y}\\ \ddot{d}_{(COM,COP),y}\end{bmatrix} = \begin{bmatrix}0 & 1\\ \omega^2 & 0\end{bmatrix}\begin{bmatrix}d_{(COM,COP),y}\\ \dot{d}_{(COM,COP),y}\end{bmatrix} + \begin{bmatrix}0\\ -\omega^2\end{bmatrix}p_{COM,y}) \quad (25)$$

Inverse kinematics allows the calculation of the coordinates of the joints, which bring the final effector to the desired position and orientation, given the absolute (operational) coordinates. When the inverse kinematic problem has a solution, it is constituted in the inverse geometric model. If an analytical solution to the inverse kinematic problem cannot be found, we can resort to numerical methods, but the disadvantage is the large volume of calculations. In the analysis of the inverse kinematics of the stepping robot, it starts with the description of the joint q with respect to the position and orientation of the final effector $x$ (the trajectory of point P of the ground) and it is defined as:

$$x = [x_1 \quad x_1 \quad \dots \quad x_n]^T \quad (26)$$

The equation of inverse kinematics is a nonlinear function because there are many joints with rotational motion, and in bipedal motion, the final effector is the moving leg, and the basis is the equilibrium [35]:

$$\dot{x} = J(q)\dot{q} \quad (27)$$

The size of $x$ is $(n \times 1)$, with $0 \le n \le 6$. In the equation, J is the Jacobi matrix, which can be defined as:

$$\mathrm{J(q)}= \begin{bmatrix}\frac{\partial x1}{\partial q1} & \cdots & \frac{\partial x1}{\partial qn}\\ \vdots & \ddots & \vdots\\ \frac{\partial xm}{\partial qn} & \cdots & \frac{\partial xm}{\partial qn}\end{bmatrix} \quad (28)$$

We will consider an optimal trajectory defined by $[q(t),\ \dot{q}(t)]$ and considering $n = m$, the speeds of the joints can be found by inverting the Jacobi matrix (that must be quadratic).

$$\dot{q} = J(q)^{-1}\dot{x} \tag{29}$$

Considering that at moment $t = 0$, we have $q(0)$, the position of the joint can be obtained as follows:

$$q(t) = \int_0^t \dot{q}\,(t)dt + q(0) \tag{30}$$

## 3. Results

The structure of the leg module consists of three components (two rotation elements and a central element) and two active articulations that move on two orthogonal axes (as in the cardan coupling). As shown in Figure 4, the articulations allow for a maximum mobility of ±90° for rotation elements 1 and 3 in relation to the central element 2. Components 1 and 3 can provide coupling with other modules through a parent system defined in the design section of the module. Each module may have a wireless transmitter and receiver with radio waves that are used for local communications and for detecting the module's position in the space of movement. The leg module has two degrees of freedom: roll (forward) and pitch (sideways). The movement is controlled at the robotic mode level by means of a microcontroller. The power supply of each robotic module can be with wires connected to the central body of robotic modules or with batteries that are located on the central element.

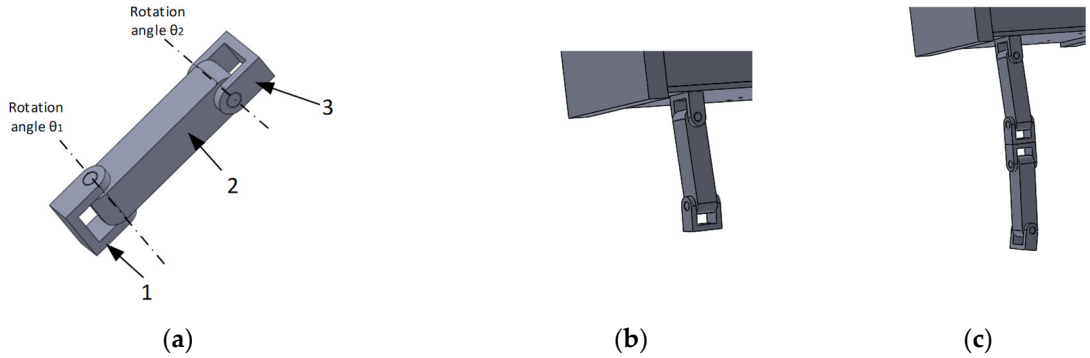
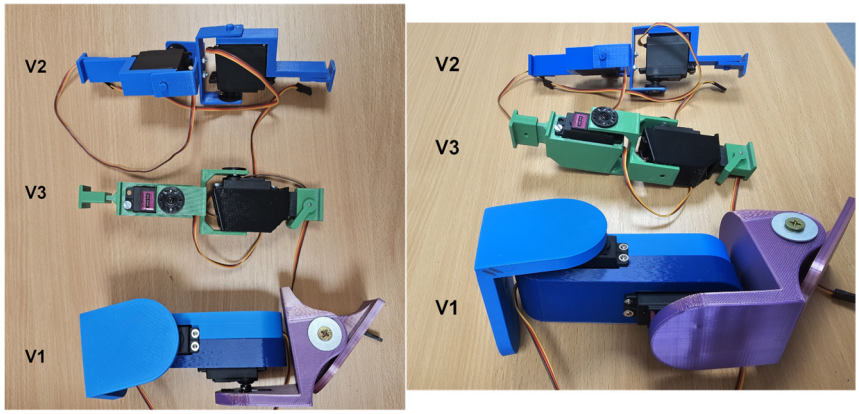



**Figure 4.** The CAD model of a module includes (**a**) a side view that focuses on the elements enabling the robot's mobility; (**b**) a leg with one leg module; and (**c**) a leg with two leg modules.

The angles are measured in local coordinate systems, with the rotation being independent of each other. A single motor can be used to drive the two lateral elements at the joints of the robotic module, or two motors can be used, one for moving each element. Through the rotation period, the resolution and speed of movement can be controlled. The displacement action can be viewed as an oscillation around the origin (at 0°) with an amplitude of 90°. The movement of the lateral movable elements of a robotic module is achieved by rotating them at the joints of the central element. A rotation element can perform a maximum rotation of ±90°.

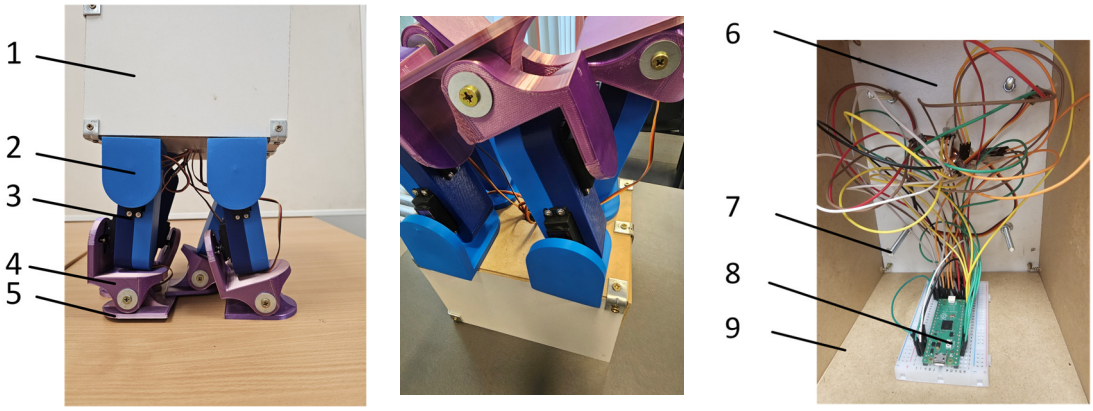
The leg of the robotic module can consist of one or more modules (see Figure 5). This reconfiguration can be carried out by the user prior to the robotic module's use (referred to as passive configuration), or it can occur while the robotic module is in motion (referred to as active configuration). Several modules designed for the feet were constructed starting from the basic model, all of which were created using 3D additive prototyping. Thus, three proof-of-concept prototypes of modular reconfigurable robots were developed.




(**a**) (**b**)

**Figure 5.** Different leg modules: (**a**) front view (version 1, version 2; version 3); (**b**) lateral view (version 1, version 2; version 3).

As can be seen in Figure 6, version 1 of the robotic module has a compact design that has space for the motors and electrical and electronic components necessary to control the module, providing rigidity and strength. One of the lateral elements of the leg module (2) is attached to the lower plate (6) by means of screws (7) to the central body of the robotic module (1). The central element of the foot (3) consists of two parts, and the side elements are fixed directly by the servo motor shaft. The other lateral element of the foot (4) connects to the actuator shaft on one side and to the ground (5) on the other. This design enhances the seating surface with the ground and incorporates a passive joint that links it to the lateral element. The controller and connection plate (8) are attached to one of the side surfaces (9). The disadvantages include high material consumption, a direct load on the servo motor shaft that limits transport capacity, and limited options for clamping the modules together. The clamping method used to secure the robotic module is through screws, which do not permit adjustments to the distances between the legs. The weight of the version 1 proof-of-concept prototype of the modular leg robot is about 1523 g.




(**a**) (**b**) (**c**)

**Figure 6.** Robotic module with leg module version 1: (**a**) front view; (**b**) up-side view; (**c**) view of the electronics inside the central body.

In version 2 (presented in Figure 7), a design was used that was as simple and flexible as possible that allows both the ease of the foot structure and the fact that the load is balanced between the joint located on the same axis as that of the servo motor (which has the possibility of being moved to make it possible to adjust the distance of the motor shaft), having space only for the motor and not for the electrical and electronic systems. Thus, the rail (2) is screwed to the lower plate (1), into which a T-element (3) can be inserted

attached to the lateral element of the foot module (4). The shape of the upper part of the central element of the foot module (4) allows the actuator (5) to be clamped, which drives one of the axes of the foot module and allows the upper part of the element to be rotated in a U shape (6) by means of a passive joint. The servo motor (5) drives the workpiece in a U shape (6), which is attached to one end of the servo motor axis. The central body of the foot consists of two U-shaped pieces (6 and 7) mounted so that the servo motors (5 and 8) act on perpendicular axes. Of course, the structure also allows the mounting of the servo motors (5 and 8) to act on parallel axes, creating a more flexible structure than that in variant 1. The actuator (8) is attached to the U-shaped element (7) and to element (8), which in turn is fixed by the T-elements (10). The foot modules can be passively clamped (before being sent for use) using rails clamped on perpendicular or parallel clamping axes. The system is fragile and allows for the connection of only 2 modules of the same type; however, the attachment to the surface of the robotic module is made using a rail, and the legs can be mounted at various distances from each other. The Version 2 prototype of the robotic leg module weighs about 771 g.

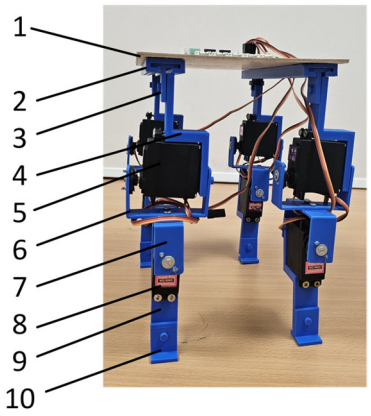



**Figure 7.** Front view of the robotic module with leg module version 2 (without the central body), the controller can be fastened on the front or rear wall surface.

In version 3, the design from variant 2 has been improved in the sense of stiffening the structure and modifying the fastening system between modules. The system is robust and allows for easy reconfiguration of the structure using passive or active intermediate elements (actuated with a servo motor). In this case we have two types of modules. A module with 2 servo motors (denoted with A in Figure 8b) that have a role only in moving the lateral elements to the central element and a module with 3 servo motors (denoted with B in Figure 8b) of which one servo motor has only the role of connecting/disconnecting the modules and the other two have a role in moving the lateral elements from the central element. Also in this case, the rail (2) is screwed to the lower plate (1) into which a T-element (4) can be inserted and fixed in the desired position by the eccentric element (3) and which is attached to the lateral element of the foot module (5) which in this case recesses the motor (6).

The central element (7) which is driven at the top by the motor (6) allows the rotation of the lateral element (5) by means of a passive joint. The central body of the foot consists of a piece composed of two U-shaped pieces glued to each other. The servo motor acting on the axis perpendicular to the upper one has the motor axis fixed by the central element (7), being embedded in the lateral element (8), which in turn is fixed by the T-elements (9). Foot modules can be passively clamped using rails clamped on perpendicular or parallel clamping axes. The system is much more solid and allows connection with a number of 4

modules of the same type and the legs can be mounted at various distances from each other from the surface of the central body (1).

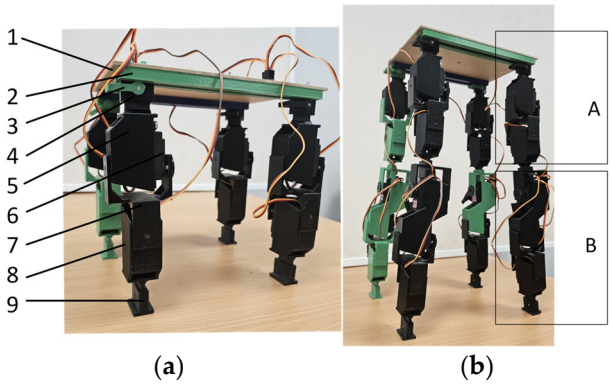



**Figure 8.** The robotic module includes leg module version 3 (without the central body), and the controller can be attached to either the front or rear surface: (**a**) with 1 leg module; (**b**) with 2 leg modules that have an active connection between them.

The type B module used allows the remotely controlled active connection or disconnection of the robotic module either from the central body board or from another module (Figure 9). The passive connection can be made at one end (being of the same type as the type A module) and the active one at the other end. The passive connection system (Figure 10a) consists of a T-element (1), which is fixed to the central element (5) by a pin that is inserted into the holes (2, 3, and 6) of the T-element (1), the fastener (3), and the side element (5), respectively. The active connection system consists of the locking element (10), which is driven by the servo motor (9), being fixed on one side by the servo motor shaft and on the other being articulated (13) with the central element (8). The connection is made by inserting element (5) of the next module into channel (11) of the central element (8) and locking with element (10) entering slot 11 and slot 7.

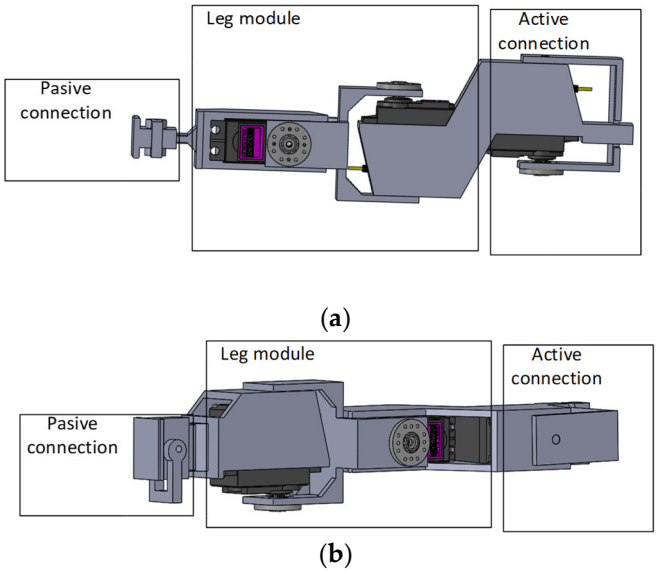



**Figure 9.** Connection mechanism of the leg robotic module version 3: (**a**) frontal view; (**b**) lateral view.

In the connected state, element (5) is fully enclosed in channel (11), so that the loads between the modules are transmitted mainly through the enclosing 3D-printed parts, while the locking element (10) only prevents the axial separation of the modules.

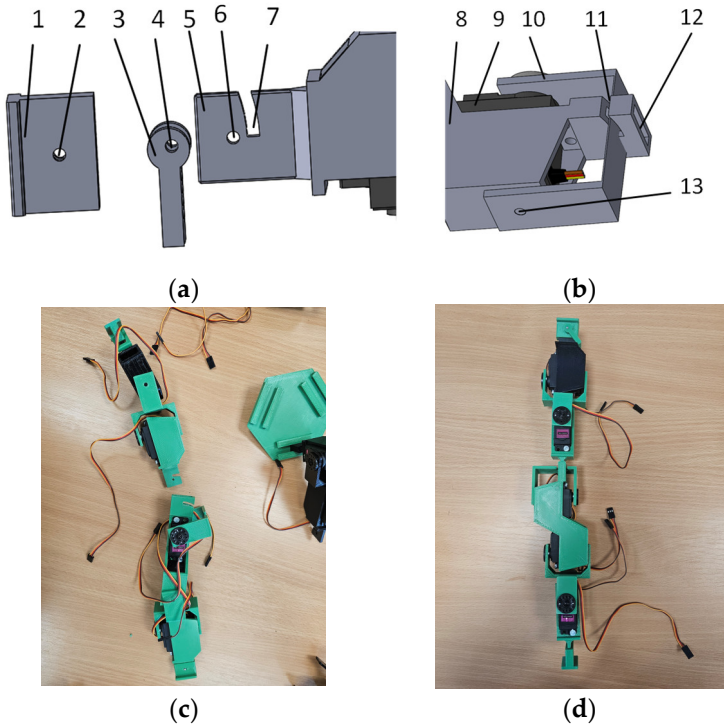



**Figure 10.** Connection mechanism of the leg robotic module version 3: (**a**) passive connection; (**b**) active connection; (**c**) two models disconnected; (**d**) two modules connected.

The version 3, type A prototype of the robotic leg module weighs approximately 1200 g, while the type B prototype weighs about 1850 g.

Using the rail that can be placed in various configurations and at various angles and the passive and active connection system, various configurations can be achieved (Figure 11). The first configuration can be used for rubble removal, the second and third configurations to overcome debris, and the fourth to secure structures, equipment, or victims. The main characteristics of the Version 3 laboratory prototype are summarized in Table 1.

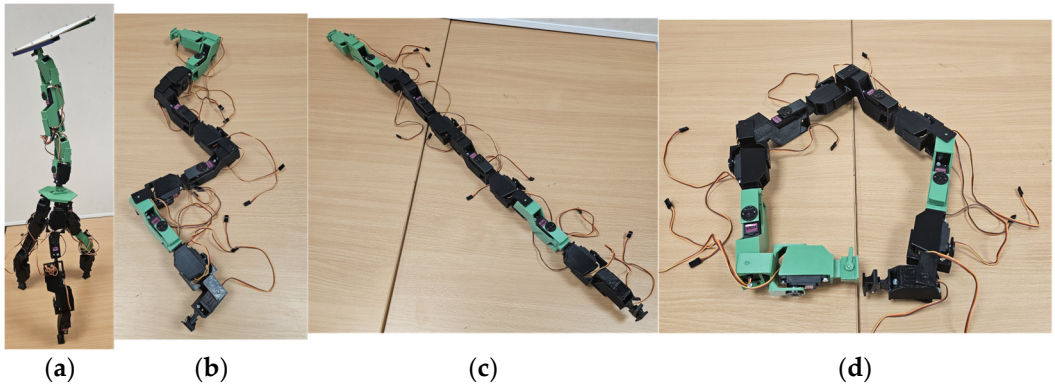



**Figure 11.** Different structures obtain by connecting several leg modules version 3 with active connecting possibility: (**a**) 1 leg device with 3 fingers fixed in a star configuration for rubble removal; (**b**) snake like robot for overcome debris; (**c**) worm like robot for overcome debris; (**d**) chain robot to secure structures, equipment, or victims.

We are able to achieve some performance parameters that we have set as our aim during the preliminary tests: a maximum obstacle overcome (L × W × H) of 120 × 120 × 50 mm, a maximum slope of 30°, a payload capacity of 1000 g, and a locomotion speed of 0.6 mm/s. When the field experimental testing is carried out in the future, these initial parameters have the potential to be significantly improved.

## 4. Discussion

### *4.1. Displacement Control of the Robotic Module*

Displacement control can be classified into several modes. The first is incremental linear displacement, in which individual modules or groups of modules move at the same speed. Another mode is displacement through sequential rotation in specified directions, with different modules considered rigid at different stages of the movement. In this case, the robot's overall configuration is the same at the start and end of the displacement. This type of movement is characteristic of chain-type robots and is particularly useful for the locomotion of long robots. Each module or group of modules rotates around the same point in the workspace, while the rotated modules may additionally perform differential translational movements. Differential translational displacement occurs when two or more legs of the robot move translationally at different speeds and in non-collinear directions. Despite the different velocities and directions of movement, the robot maintains the same overall configuration at both the beginning and the end of the displacement. Trajectory planning and navigation planning are also parts of displacement control. They decide the robot's desired path and movement strategy. Based on complexity, displacement can be classified as simple or compound. Simple displacement represents a fundamental type of movement that cannot be further decomposed into other displacement types. Compound displacement, in contrast, is formed by combining two or more simple displacement modes. Compound displacement can be achieved in three fundamental ways: through joints, hierarchically, or morphologically. Joint-based displacement is generated through the coordinated movement of the robot's joints, hierarchical displacement is organized across different structural or functional levels of the robot, and morphological displacement is achieved through changes in the robot's configuration or morphology.

Some examples of simple movement are wheeled or leg movements. The two types of movement can be combined, resulting in a complex movement such as the movement of a man on wheels. Complex displacement can be achieved by connecting multiple modules or groups of modules with joints. In mobile robots, joints are active links between different modules that may not have the same form of locomotion. The joints do not necessarily form chain-type robots, but also robots in the plane. An advantage of this way of combining trips is that you can obtain any type of trip. Thus, two modules can be connected through a connection with six degrees of freedom without affecting locomotion. Another advantage is that the connection area requires a small width compared to the length of the modules, so the robot can move in areas with high displacement constraints. The width allows the robot to move through obstacles, and the length allows it to pass over large obstacles. The load capacity of articulated robots is higher than in the case of other robots [36].

A hierarchical structure displacement involves the combination of two types of displacements, superior and lower. The upper displacement is executed at an intermediate level so that only the group of modules executing the lower displacement has a point of contact with the ground. Typically, the lower displacement is implemented at a lower level than the upper displacement and is executed at much higher frequencies. Any type of displacement can be superior or inferior. Although joint displacements are quite common in nature, hierarchical combinations are rare. The advantage of this type of movement lies in the fact that the simple movements that compose it do not influence the efficiency or speed of the movement, such as articulated movements. This is because displacement is additive.

A morphologically compound displacement is defined by connecting simple displacements in the sense that each of the displacements acts on a different axis. The elements that perform simple displacements can touch the ground and cannot be separated

by removing the joint. A simple example is moving on a sphere by turning and rolling a wheel in any direction. Thus, this movement can be viewed as a morphological combination of the movements produced by an infinite number of wheels. Unlike the other modes of displacement, not every type of displacement can be combined. A disadvantage of this system is that generating movement is complex; however, an advantage is that it allows for movement in multiple directions simultaneously. When moving a robot with a stable shape, the movement is achieved by adjusting the joints. An alternative to controlling movement is mimicking the modes of movement found in the living world.

In this paper we will present centralized control of the robot via a trajectory control table (see Table 2). To implement this type of displacement, each module is stored in the controller's memory in simple modes of displacement. Centralized control controls the transition between the different modes, using a displacement control table depending on the shape and action of the robot. Each column in the table corresponds to the simple way of moving for all modules for each cycle of movement. Therefore, each module receives the information from its column in the displacement table and executes the commanded action, using a closed-loop control system. Because the control of the robot is done remotely, we consider that a sequence of actions must be simulated graphically for good feedback, and the sequence of movements is saved in a control file. A central controller uses this file to play the sequence of actions.

After the validation of the movement, the trajectory will be generated, and the movement of each articulation will be completely specified using a tabular representation. The table rows show the rotation torque of robot modules in degrees. Each column of the table will show kinematic coupling movements for multiple stages. Motion control begins at the first step in the table and concludes when the rotating couplers reach their intended position. After finishing the last step on the table, step 0 is resumed.

**Table 2.** Absolute values of the rotations in couplings (example of values for line forward movement).

| Step | Rotations in Couplings | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | Leg 1 | | Leg 2 | | Leg 3 | | Leg 4 | |
| | M1 | M2 | M3 | M4 | M5 | M6 | M7 | M8 |
| 0 | 0° | 0° | 0° | 0° | 0° | 0° | 0° | 0° |
| 1 | −20° | 0° | −20° | 0° | −20° | 0° | −20° | 0° |
| 2 | 0° | 0° | −20° | 0° | −20° | 0° | −20° | 0° |
| 3 | 0° | 20° | −20° | 0° | −20° | 0° | −20° | 0° |
| 4 | −20° | 20° | −20° | 0° | −20° | 0° | −20° | 0° |
| 5 | −20° | 20° | 0° | 0° | −20° | 0° | −20° | 0° |
| 6 | −20° | 20° | 0° | 20° | −20° | 0° | −20° | 0° |
| 7 | −20° | 20° | −20° | 20° | −20° | 0° | −20° | 0° |
| 8 | −20° | 20° | −20° | 20° | 0° | 0° | −20° | 0° |
| 9 | −20° | 20° | −20° | 20° | 0° | 20° | −20° | 0° |
| 10 | −20° | 20° | −20° | 20° | −20° | 20° | −20° | 0° |
| 11 | −20° | 20° | −20° | 20° | −20° | 20° | 0° | 0° |
| 12 | −20° | 20° | −20° | 20° | −20° | 20° | 0° | 20° |
| 13 | −20° | 20° | −20° | 20° | −20° | 20° | −20° | 20° |
| 14 | 0° | 20° | 0° | 20° | 0° | 20° | 0° | 20° |

Tabular movement is not designed to handle configuration changes in such a structure, which is its only drawback. The motion table must be altered whenever the configuration changes, no matter how slight, because the robot structure's module order no longer matches the table. When two robots of this type combine to construct a structure,

a new motion table is needed. Concatenating the motion tables of the two robots may misalign the steps. On rough, unprepared terrain, robots' rotational torques cannot be determined from the start.

We use status messages to depict a movement by observing that the motion table resembles the component modules' motions. The action module m must be done at step t, and the movement module (m − 1) must perform at step (t + 1). Instead of storing the whole motion control table, each module represents and distributes this displacement as a sequence of drive motor actions. If one module makes this meandering movement, it must choose and execute one of the above motions to match its movements and those of the nearby modules. A status message is used to coordinate the movement of numerous modules via the robotic structure and allows each module to inform its neighbor of the tasks it must complete and to disseminate the status messages. Because status messages can alter throughout propagation, they differ from broadcast messages. To implement status message principle-based robot system control, each status message must react to other status messages received from local operations. Controlling sensors and effectors, updating system statuses, changing status messages, and creating new ones are examples. Each module decides what to do based on status messages, structural topology, and general information.

Overall, this protocol has several important characteristics [20]. It is a distributed and fault-tolerant approach in which no central module is assigned the role of system master, meaning that any module can assume this function. Consequently, the failure of a single module does not necessarily compromise the operation of the entire system. The protocol also supports cooperation among modules, allowing them to determine the behavior of their neighbors based on local and topological information without requiring each module to have a unique identity. Coordination is asynchronous, as system operation does not depend on a real-time clock; instead, status messages exchanged along the module chain are used to coordinate signals and actions. Finally, the protocol is scalable because its control mechanism can accommodate changes in the robot's topology. Modules can be added, removed, or reordered relative to the initial configuration without requiring a fundamental modification of the control mechanism. These properties have not yet been verified on the present prototype, which uses the centralized motion table described above.

### *4.2. Features of the Drive System Command and Control System*

Reconfigurable robots use sensors to monitor the status of their modules and perceive the surrounding environment. Internal sensors mainly measure the position and displacement of the kinematic couplings and the accelerations that define the modules' orientation relative to gravity. External sensors perform several important functions during robot operation. They determine the position and orientation of the coupling faces required for connecting and disconnecting modules, assist in selecting the appropriate means of locomotion, and enable the robot to respond to changing environmental conditions by triggering specific behaviors. In addition, external sensors are used to detect obstacles and other environmental barriers, allowing the robot to identify suitable strategies for overcoming them.

In our work, acceleration sensors are used to measure the inclination or slope of the central body and the angle of rotation around its axes. They also enable the detection of movement or rest conditions, provide information about shocks or collisions, and record module accelerations along one or more axes, including both dynamic accelerations caused by vibrations and static accelerations resulting from gravity. External sensors also play an important role in the operation of reconfigurable robots. For testing purposes, we use ultrasonic sensors because they provide effective, precise, and non-contact distance measurements to objects in the surrounding environment. These sensors can be used to

identify the position and orientation of additional modules for potential attachment and connection, as well as to detect obstacles in the robot's surroundings and determine their position and configuration. This information allows the robot to develop an appropriate strategy for avoiding obstacles and responding to them in real time. In addition, ultrasonic sensors can measure the distance to fixed or moving markers, providing information that can be used to determine the robot's position within the workspace. The sensor identifies an object and measures its distance by generating a brief pulse at 40 KHz (ultrasonic range) at 340 m/s, moving at the speed of sound in the air, and waiting for the obstacle to reflect the waves.

Recent advances include digital servo motors with significant functional advantages over traditional ones. Digital servos are identical to normal servo motors, but a computer analyzes input data and controls the motor. Differences in input signal processing and servo motor power transmission reduce downtime, increase resolution, and increase holding torque. Traditional servo motors are not powered at idle.

For driving robotic system modules, servos are reliable, and in our project we use MG996R. Digital servo motors supply voltage when a start command is received or torque is applied to the output shaft. This power, the highest voltage, is pulsed on/off at 50 cycles per second to generate small voltage pulses. By increasing pulse duration, a speed control is generated until the motor is under maximum strain, accelerating the servo arm to the new position. The servo's electronics receive the end of the stroke signal and shorten the power pulses until there is no voltage and the motor stops. The main advantages of digital servo motors arise from their integrated microprocessor, which processes the input signals and applies the required settings before transmitting them to the servo motor. Optimizing the servo's performance involves programming the microprocessor to meet the desired functions and operating requirements. This enables precise adjustments to the pulse duration and the voltage supplied to activate the motor, leading to enhanced control and responsiveness.

Another important characteristic of digital servo motors is their higher pulse frequency. While conventional servo motors typically receive approximately 50 pulses per second, digital servo motors can receive around 300 pulses per second. As a result, the voltage applied to the motor is updated more frequently, allowing the motor to respond more rapidly and achieve greater movement precision. Although the pulse duration may decrease as the frequency increases, the higher number of pulses enables more frequent adjustments to the motor position. Servo motors have unique features that make them ideal for driving pulleys and wheels with a 180° rotation angle. The engine can receive speed, direction, and final position targets with one command signal.

Raspberry Pi-compatible ARM Cortex-based microcontrollers will power this robot. The first microcontroller is the MAKER-PI-RP2040 (Cytron Technologies Sdn Bhd (Penang HQ), Adress: Lorong Industri Impian 1, Taman Industri Impian, 14000 Bukit Mertajam, Pulau Pinang, Malaysia), which uses the RP2040 microcontroller. A two-channel DC motor driver, 4 servo motor ports, and 7 I/O connectors are on this board. The platform supports popular MicroPython and C/C++. The board contains a built-in LiPo/Li-ion battery with 1 cell (overcharged and over-discharged protection), an on/off switch, 13× GPIO pin status indication LEDs, and 1× piezo buzzer with a mute button. It supports automated power selection from USB 5V, LiPo (1 cell), or Vin.

### *4.3. Module Robot Control in a VR Environment*

In this paragraph, we will demonstrate an example of the command that may be used for the robot that is being shown (Figure 12). This command has the potential to be significantly enhanced during the process of developing its control. It is based on the concept that the virtual reality environment is utilized not as a real-time digital twin but rather as

a visualization and teleoperation interface. The motion sequence is generated and stabilized in the virtual model before being sent to the robot, and no feedback from the physical robot is used in the virtual model. The robotic module has been designed for remote control, allowing the user to connect at any time through a PC, either by using a keyboard or a haptic device. In order to build the model in the virtual reality environment, the CAD model of the robot made in Solidworks was used, which was exported as STL and imported into the Blender 4.4.3 program. Based on the kinematics of the mobile robot, a kinematic chain named "skeleton" is built from several segments that can rotate relative to each other. With the help of the skeleton, we can control the movements of the different component parts.

The model will be saved in the FBX format, which will then be imported into Unreal software (Unreal Engine 5 version 5.8. In Unreal software, a component will be created to animate the character by adding controls that follow the locomotion algorithm. According to the keyboard commands, the model moves in real time in the VR environment. To retrieve the data from the glove, we use the plugin that communicates with the glove through the serial port and transmits the data in the virtual environment using the Serial Communication Plugin for Unreal Engine.

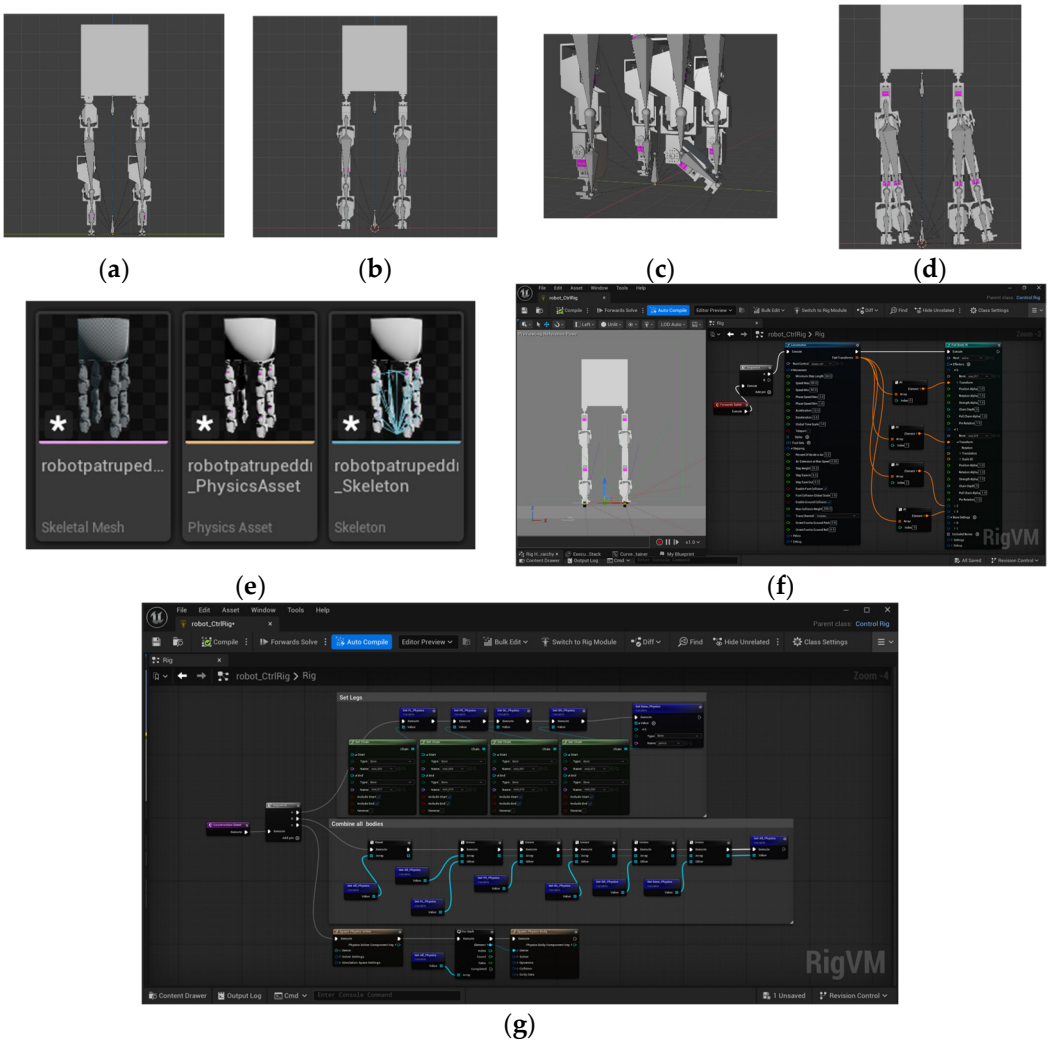



**Figure 12.** Building the model to be imported into the Unreal software: (**a**) STL model is imported in Blender software, (**b**) skeleton build according to robot kinetics, (**c**) skeleton segments can move relative to each other, (**d**) legs modules move according to the calculated control parameters, (**e**) the components of the imported model in Unreal, (**f**) the control sequence of the module is built using

locomotor inverse kinematics, (**g**) the control sequence of the module is built using locomotor and calculating the movement of each element.

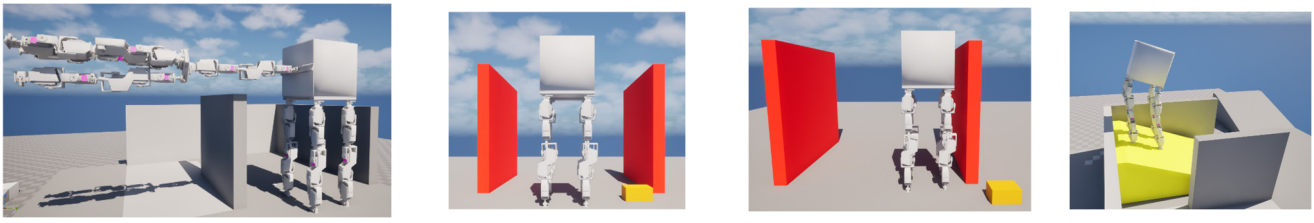
The control of characters within the virtual environment, which in our case includes the robot model and the surrounding walls, is based on several essential components. The first component is the skeletal mesh, which represents the visible body of the robot and corresponds to the actual 3D model, including its textures and visual details such as the casing, legs, and screws. The skeletal mesh is the component that is displayed to the operator on the screen and provides the visual representation of the robot within the virtual environment. The second component is the physics asset, which represents the robot's physical body and is used to detect interactions and collisions with other elements of the environment. It defines invisible geometric shapes, such as capsules, that surround the robot and allow the graphics and physics engine to determine how the robot interacts with its surroundings. This component prevents the robot from passing through walls, floors, or other physical objects and ensures a more realistic simulation of its movement. The third component is the skeleton, which represents the robot's kinematic structure. It forms an invisible internal framework that connects the different parts of the robot and defines their relationships and movements. Although the skeleton is not directly visible in the virtual environment, it is essential for controlling the robot's joints and animating its individual components.

To control the robot in the virtual environment by moving a reference point located at its base, a Control Rig must be implemented to define the kinematics of each movable component represented as an element of the robot's skeleton. The kinematic structure is organized as a tree and relies on three main functions: the Locomotor, the nodes that distribute the generated positions, and the Full Body Inverse Kinematics system.

The locomotor functions as a reference point whose displacement generates the trajectory of the robot's moving elements according to the defined kinematic constraints. It calculates the inverse kinematics based on the trajectory of the selected reference point and takes into account parameters such as velocity, acceleration, stride height, stride length, and ground contact or collision. As a result, it generates a sequence of possible foot positions in the workspace. The nodes receive the list of positions generated by the locomotor and distribute them among the individual robot legs. Each position is assigned to a specific leg according to its index, with Index 0 corresponding to Leg 1, Index 1 to Leg 2, and so on. The nodes then transmit the corresponding coordinates to the inverse kinematics system for further processing. The Full Body Inverse Kinematics system receives the desired positions generated for each leg and adjusts the robot's joints so that the feet reach the specified target points on the ground. It controls the movement of the entire body, starting from the root or central body, and automatically adjusts the intermediate joints to produce coordinated and natural movements. This prevents the robot from appearing excessively rigid or mechanically disconnected during locomotion. In conclusion, the locomotor determines where the robot should place its feet on the ground, the nodes distribute the corresponding instructions to each leg, and the full-body inverse kinematics system adjusts the robot's joints so that the feet accurately reach the defined target positions.

The second control variant consists of allowing the implementation of a more complex model through which each component element of the foot moves separately, then combines all bodies (creating a single complete list of all the moving elements of the robot), takes the list of elements together with its movements, and activates the physics/dynamic simulation in the graphics engine. This code runs only once at the beginning to record the structure of the robot and activate the movement of each element.

The Sequence node divides the action into two consecutive steps that are executed in a strict order. During the first step, the positioning stage, the Locomotor Node determines where the robot should place its feet, while the Full Body IK system adjusts the robot's legs and positions the feet on the ground according to the calculated targets. Immediately afterwards, the second step, the physics stage, is initiated by the Step Physics Solver node, which applies gravity, weight, and collision effects to the robot's already positioned bones. In this way, the robot first places its feet on the ground and then transfers its weight onto them through the physics engine, resulting in a more realistic and stable movement (Figure 13).



(**a**) (**b**) (**c**) (**d**)

**Figure 13.** Moving the robot in the virtual environment: (**a**) adding different components to the module robot; (**b**,**c**) movement of walls according to the sensors parameters; (**d**) navigating through obstacles.

Using this virtual model the user can control in a simulated world, can control various elements of the robot regardless of its structure, can visualize obstacles that can be modeled as movable walls approaching the robot according to the values provided by ultrasonic sensors.

## 5. Conclusions

Compared with conventional rigid, single-unit rescue platforms primarily intended for exploration, the proposed system integrates locomotion, communication, power supply, and material delivery within a modular and adaptable architecture. Its configuration can be modified according to terrain conditions and mission requirements, extending its functionality from environmental exploration to direct support of trapped victims.

This paper presented the comprehensive design, unified kinematic modeling, stability analysis, and multi-stage implementation of an innovative modular reconfigurable robotic system designed for search and rescue operations in post-disaster environments. Unlike existing rigid single-unit robotic platforms that face severe limitations in terrain adaptability and power autonomy, the proposed multi-module architecture introduces a flexible paradigm driven by a multifunctional structural tube.

A major theoretical contribution of this research is the symbolic unification of kinematic and structural models across both wheeled and legged locomotion systems. By establishing an equivalent radius parameter, the discrete multi-phase biomimetic gait cycle was mapped into continuous state equations identical to differential-drive kinematics. The analytical formulation exposed the coupled, non-linear nature of positioning deviations, specifically isolating longitudinal, orientation, and lateral slip errors. To mitigate these tracking errors at low operational velocities, a simplified linear control law was implemented using an experimental parameter and a command variable. This solution is intended to reduce real-time computational overhead on the integrated microcontroller while preserving trajectory stability at low speeds.

The stability analysis of the robot systematically addressed static, quasi-static, and dynamic equilibrium using core criteria such as the center of mass projection, Center of

Pressure (CoP), and Zero-Moment Point (ZMP). Incorporating the accelerations and moments of inertia of the robot's rigid components, Huang's equations were derived to determine the precise tracking of the ZMP coordinates. To ensure dynamic stability over fragmented terrain, a Linear Inverted Pendulum Model (LIPM) was applied, leveraging the structural natural frequency to predictively regulate center of mass coordinates relative to foot-ground contact projections.

From an experimental perspective, the physical structure was validated through 3D additive prototyping across three successive design iterations. Version 3 optimized mechanical stiffness and connection flexibility by introducing specialized Type A modules for localized joint locomotion and Type B modules equipped with an active servo-driven mechanism for remote connection and disconnection.

At the displacement control level, two strategies will be evaluated comparatively in the future: a centralized method utilizing kinematic rotation tables—highly efficient for predefined paths but rigid against topological changes—and a decentralized protocol based on the asynchronous propagation of status messages between modules. The latter is expected to provide scalability, allowing modules to be dynamically added, removed, or reordered without modifying the core control mechanism.

The iterative development of the leg module led from the compact and rigid V1 architecture, characterized by screw-based connections and limited adjustability, to the more flexible but mechanically fragile V2 configuration, which allowed only two modules of the same type to be connected. The final V3 design increased structural rigidity and introduced remotely actuated connection and disconnection, while allowing up to four modules of the same type to be interconnected and assembled into different configurations, including star-, snake-, worm-, and chain-like structures.

By representing legged locomotion through an equivalent virtual-wheel model, a common kinematic formulation can be applied to both wheeled and legged module configurations under the adopted simplifying assumptions. This approach reduces the need for completely independent locomotion models and provides a common basis for the control of a reconfigurable platform combining different locomotion mechanisms.

The proposed status-message-based control protocol provides a distributed, fault-tolerant, asynchronous, and scalable solution for coordinating reconfigurable modules, allowing the control architecture to accommodate changes in system topology. In parallel, the VR environment enables remote human–robot interaction through keyboard or haptic control and provides a safe platform for preliminary evaluation of locomotion and control algorithms before experimental implementation under realistic operating conditions.

The results of this study, together with the trends identified in the analyzed literature, support the perspective that modular reconfigurable robots represent a promising direction for future search-and-rescue systems. Their ability to combine different locomotion, sensing, communication, and support functions within a single adaptable architecture can facilitate operation in environments characterized by variable terrain and changing mission requirements. In this context, further research should focus on improving reconfiguration strategies, locomotion and stability control, inter-module coordination, and experimental validation under realistic disaster conditions, with the aim of extending the applicability and operational autonomy of such systems.

Future research efforts will focus on the experimental calibration of the coefficient on the physical prototype and conducting extensive field testing within simulated large-scale structural collapse scenarios to evaluate the operational boundaries of this hybrid locomotion platform under real-world rescue constraints.

**Author Contributions:** Conceptualization, A.D.A. and C.G.A.; methodology, C.G.A. and R.C.; software, M.-V.G. and A.D.A.; investigation, V.A.S., M.-V.G. and L.B.; resources, V.A.S. and L.B.; writing—review and editing, R.C. and A.D.A.; supervision, C.G.A. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Data Availability Statement:** Data are contained within the article.

**Acknowledgments:** The authors have reviewed and edited the output and take full responsibility for the content of this publication.

**Conflicts of Interest:** The authors declare no conflicts of interest.

## Abbreviations

The following abbreviations are used in this manuscript:

| | |
|---|---|
| CoM | Center of Gravity |
| CoP | Center of Pressure |
| IMU | Inertial Measurement Unit |
| LIPM | Linear Inverted Pendulum Model |
| ZMP | Zero-Moment Point |
| FZMP | Zero-Moment Point with Friction |
| VR | Virtual Reality |

## References

1. Liu, J.; Wang, Y.; Li, B.; Ma, S. Current research, key performances and future development of search and rescue robots. *Front. Mech. Eng. China* **2007**, *2*, 404–416.
2. Bogue, R. Search and rescue and disaster relief robots: Has their time finally come? *Ind. Robot* **2016**, *43*, 138–143.
3. Thanu Thavasi, P.; Suriyakala, C.D. Sensors and Tracking methods used in Wireless Sensor Network Based Unmanned Search and Rescue system—A Review. *Procedia Eng.* **2012**, *38*, 1935–1945.
4. Henkel, Z.; Suarez, J.; Srinivasan, V.; Murphy, R.R. Medical Field Exercise With a Social Telepresence Robot. *J. Behav. Robot. Paladyn* **2016**, *7*, 1–14.
5. Bezzini, R.; Fritsch, S.; Bassani, G.; Avizzano, C.A.; Filippeschi, A. A Novel Compact Rolling Element Eccentric Planetary Gearbox Design for Lightweight and Backdrivable Wearable Robots Actuators. *Robotics* **2026**, *15*, 162.
6. Bezzini, R.; Bassani, G.; Avizzano, C.A.; Filippeschi, A. Nested Planetary-Cycloidal Transmissions Design for Compact and Lightweight Exoskeletons Actuators. *Rep. Mech. Eng.* **2026**, *7*, 253–272.
7. Tagliavini, L.; Colucci, G.; Botta, A. Wheeled Mobile Robots: State of the Art Overview and Kinematic Comparison Among Three Omnidirectional Locomotion Strategies. *J. Intell. Robot Syst.* **2022**, *106*, 57.
8. Taheri, H.; Zhao, C.X. Omnidirectional mobile robots, mechanisms and navigation approaches. *Mech. Mach. Theory* **2020**, *153*, 103958.
9. Majithia, A.; Shah, D.; Dave, J.; Kumar, A.; Rathee, S.; Dogra, N.; Chiniwar, V.H.M.; Hiremath, D.S. Design, motions, capabilities, and applications of quadruped robots: A comprehensive review. *Front. Mech. Eng.* **2024**, *10*, 1448681.
10. Roșu, R.A.; Prisăcariu, E.G.; Dumitrescu, O. Integration Challenges of Turbine-Powered UAVs: Thermal, Structural, Acoustic, and Operational Perspectives. *Technologies* **2026**, *14*, 455.
11. Mohd Daud, S.M.S.; Mohd Yusof, M.Y.P.; Heo, C.C.; Khoo, L.S.; Singh, M.K.C.; Mahmood, M.S.; Nawawi, H. Applications of drone in disaster management: A scoping review. *Sci. Justice* **2022**, *62*, 30–42.
12. Recchiuto, C.T.; Sgorbissa, A. Post-disaster assessment with unmanned aerial vehicles: A survey on practical implementations and research approaches. *J. Field Robot.* **2018**, *35*, 459–490.
13. Prisăcariu, E.G.; Roșu, R.A.; Dumitrescu, O.; Ciobanu, R.R. A Review of Micro Gas Engines for UAV Propulsion: Fundamentals and Emerging Technologies. *Drones* **2026**, *10*, 543.
14. Zou, W.; Ma, E.; Chen, H.; Lin, F.; Xiao, Z.; Zhang, Z.; Jiang, J.; Niu, Y.; Yu, P.; Wang, C. Locomotion for Insect-Scale Robots With Bionic Strategies: A Review. *J. Field Robot.* **2025**, *42*, 1586–1616.
15. Transeth, A.A.; Pettersen, K.Y.; Liljebäck, P. A survey on snake robot modeling and locomotion. *Robotica* **2009**, *27*, 999–1015.

16. Yim, M.; Shen, W.-M.; Salemi, B.; Rus, D.; Moll, M.; Lipson, H.; Klavins, E.; Chirikjian, G.S. Modular Self-Reconfigurable Robot Systems [Grand Challenges of Robotics]. *IEEE Robot. Autom. Mag.* **2007**, *14*, 43–52. https://doi.org/10.1109/MRA.2007.339623.
17. Seo, J.; Paik, J.; Yim, M. Modular Reconfigurable Robotics. *Annu. Rev. Control Robot. Auton. Syst.* **2019**, *2*, 63–88. https://doi.org/10.1146/annurev-control-053018-023834.
18. Brunete, A.; Ranganath, A.; Segovia, S.; Perez de Frutos, J.; Hernando, M.; Gambao, E. Current Trends in Reconfigurable Modular Robots Design. *Int. J. Adv. Robot. Syst.* **2017**, *14*, 1729881417710457. https://doi.org/10.1177/1729881417710457.
19. Yim, M.; Duff, D.G.; Roufas, K.D. PolyBot: A Modular Reconfigurable Robot. In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), San Francisco, CA, USA, 24–28 April 2000; Volume 1, pp. 514–520. https://doi.org/10.1109/ROBOT.2000.844106.
20. Castano, A.; Behar, A.; Will, P.M. The Conro Modules for Reconfigurable Robots. *IEEE/ASME Trans. Mechatron.* **2002**, *7*, 403–409. https://doi.org/10.1109/TMECH.2002.806233.
21. Murata, S.; Yoshida, E.; Kamimura, A.; Kurokawa, H.; Tomita, K.; Kokaji, S. M-TRAN: Self-Reconfigurable Modular Robotic System. *IEEE/ASME Trans. Mechatron.* **2002**, *7*, 431–441. https://doi.org/10.1109/TMECH.2002.806220.
22. Salemi, B.; Moll, M.; Shen, W.-M. SUPERBOT: A Deployable, Multi-Functional, and Modular Self-Reconfigurable Robotic System. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Beijing, China, 9–15 October 2006; pp. 3636–3641. https://doi.org/10.1109/IROS.2006.281719.
23. Davey, J.; Kwok, N.; Yim, M. Emulating Self-Reconfigurable Robots—Design of the SMORES System. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vilamoura, Portugal, 7–12 October 2012; pp. 4464–4469. https://doi.org/10.1109/IROS.2012.6385845.
24. Kamegawa, T.; Yamasaki, T.; Igarashi, H.; Matsuno, F. Development of the Snake-Like Rescue Robot "KOHGA". In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), New Orleans, LA, USA, 26 April–1 May 2004; pp. 5081–5086. https://doi.org/10.1109/ROBOT.2004.1302523.
25. Arai, M.; Tanaka, Y.; Hirose, S.; Kuwahara, H.; Tsukui, S. Development of "Souryu-IV" and "Souryu-V": Serially Connected Crawler Vehicles for In-Rubble Searching Operations. *J. Field Robot.* **2008**, *25*, 31–65. https://doi.org/10.1002/rob.20229.
26. Wright, C.; Buchan, A.; Brown, B.; Geist, J.; Schwerin, M.; Rollinson, D.; Tesch, M.; Choset, H. Design and Architecture of the Unified Modular Snake Robot. In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), Saint Paul, MN, USA, 14–18 May 2012; pp. 4347–4354. https://doi.org/10.1109/ICRA.2012.6225255.
27. Whitman, J.; Zevallos, N.; Travers, M.; Choset, H. Snake Robot Urban Search After the 2017 Mexico City Earthquake. In Proceedings of the IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR), Philadelphia, PA, USA, 6–8 August 2018. https://doi.org/10.1109/SSRR.2018.8468633.
28. Hatazaki, K.; Konyo, M.; Isaki, K.; Tadokoro, S.; Takemura, F. Active Scope Camera for Urban Search and Rescue. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), San Diego, CA, USA, 29 October–2 November 2007; pp. 2596–2602. https://doi.org/10.1109/IROS.2007.4399386.
29. Spong, M.W.; Hutchinson, S.; Vidyasagar, M. *Robot Modeling and Control*; John Wiley & Sons Inc.: Hoboken, NJ, USA, 2006.
30. Huang, J.; Zhou, J.; Wang, Z.; Law, J.; Cao, H.; Li, Y.; Wang, H.; Liu, Y. Modular Origami Soft Robot with the Perception of Interaction Force and Body Configuration. *Adv. Intell. Syst.* **2022**, *4*, 2200081.
31. Liu, Y.; Nejat, G. Robotic Urban Search and Rescue: A Survey from the Control Perspective. *J. Intell. Robot. Syst.* **2013**, *72*, 147–165. https://doi.org/10.1007/s10846-013-9822-x.
32. Delmerico, J.; Mintchev, S.; Giusti, A.; Gromov, B.; Melo, K.; Horvat, T.; Cadena, C.; Hutter, M.; Ijspeert, A.; Floreano, D.; et al. The Current State and Future Outlook of Rescue Robotics. *J. Field Robot.* **2019**, *36*, 1171–1191. https://doi.org/10.1002/rob.21887.
33. Armah, S.; Yi, S.; Taher, A.-L. Implementation of autonomous navigation algorithms on two-wheeled ground mobile robot. *Am. J. Eng. Appl. Sci.* **2014**, *7*, 149–164.
34. Kasaei, M.; Lau, N.; Pereira, A. Comparison Study of Well-Known Inverted Pendulum Models for Balance Recovery in Humanoid Robot. In Proceedings of the MAP-i 2019, Braga, Portugal, 12 September 2019.
35. Sobotka, M.; Wollherr, D.; Buss, M. A Jacobian Method for Online Modification of Precalculated Gait Trajectories. In Proceedings of the 6th International Conference on Climbing and Walking Robots, Catania, Italy, 17–19 September 2003; pp. 435–442.
36. Yamada, S.; Hirose, S.; Endo, G.; Nabae, H. R-CRANK: Amphibious all terrain mobile robot. In *Proceedings of the 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2016)*; IEEE: Piscataway, NJ, USA, 2016; pp. 1067–1072.

**Disclaimer/Publisher's Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

https://doi.org/10.3390/app162110915