Research paper

Design and adaptive control of a kinematically redundant robot with enhanced trajectory tracking for climbing in tight spaces

Mohammad Adinehvand [a],[*], Ehsan Asadi [a], Chow Yin Lai [b], Hamid Khayyam [a], Reza Hoseinnezhad [a]

[a] School of Engineering, RMIT University, Melbourne, VIC 3083, Australia
[b] UCL, Gower St, London WC1E 6BT, UK

A B S T R A C T

This work investigates the design and adaptive control of a miniature robot with multi-modal locomotion which has the ability to climb inside train bogies for inspection purposes. We propose and analyse a kinematically redundant mechanism with six 2-DOF couple joints. The robot can squeeze through narrow spaces and also climb on surfaces with transitions, irregularities and discontinuities. The unique design allows desirable self-motion close to obstacles but imposes strict requirements in motion control and precise path following. This paper applies such redundancy and self-motion by constructing an adaptive controller with time-varying safety constraints for all twelve joints of the mechanism. The control strategy relies on the time-varying Barrier Lyapunov Function to bound the trajectory error. It also deploys an adaptive radial basis function neural network to estimate the system parameters of the robot. Various simulation experiments show that the proposed controller satisfies all safety and physical joint constraints. It also minimises trajectory tracking error irrespective of initial conditions, disturbances, and unmodelled dynamic effects. Finally, we compare the tracking results with those obtained by a Feedback Linearisation controller and a Quadratic Lyapunov Function-based controller. Results demonstrate enhanced locomotion and trajectory tracking for collision-free manoeuvring in tight spaces.

1. Introduction

Robotics-enabled inspection in the rolling stock industry offers several advantages, including reduced operation, capital costs, waste, labour turnover, and saving on space and time. One of the most challenging areas of the train for automated inspection is bogie, which is a confined space underneath the train consisting of the frames, wheelset, suspensions, traction linkage assembly, anti-roll bar assembly, traction-motor and gearbox. The bogie environment is very tight and unstructured, with surface transitions, irregularities, and discontinuities that are hard to traverse by most known climbing mechanisms. Wall-climbing robots have employed various mechanisms to achieve continuous, step-by-step, and hybrid locomotion. In addition, they utilise different adhesive technologies such as magnetic force, mechanical systems, pneumatics, chemical, and electrostatic adhesion to overcome the gravity force.

Continuous-motion robots, such as wheeled mobile robots (WMRs) and tracked mobile robots (TMRs), are generally fast and simple in both hardware and software requirements. MagneBike [1] is well-discussed magnet-wheeled robot. This robot is capable

Table 1

Comparison of relevant climbing robots using continuous-motion, step-by-step and hybrid locomotion.

Requirement	Continuous robots		Step by step robots			Hybrid robots
Locomotion	Wheel [1]	Track [9]	Chain [4]	Chain [5]	Brachiation [6]	Hybrid [7]
Adhesive	Magnet	Magnet	Pneumatic	Pneumatic	Mechanical	Magnet
Size[a]	$185 \times 131 \times 153$	$430 \times 145 \times 234$	$45 \times 45 \times 248$	1260	$\sim 240 \times 165$	$628 \times 130 \times 38$
Mass (g)	670	7500	335	16,100	1288	4000
Speed	~ 17 cm/s	4–50 cm/s	N.A[c]	36.66 cm/s	~ 3.27 cm/s	N.A
Transition[b]	$\approx$	✗[f]	✓[e]	✓	$\approx$[d]	✓
Irregularities[b]	✗	✓	✓	✓	✗	$\approx$
Discontinuities[b]	✗	✓	✓	✓	✓	$\approx$
Obstacle[b]	✗	$\approx$	$\approx$	$\approx$	✗	$\approx$

[a]All dimension in mm.

[b]Environment challenges.

[c]Not Available.

[d]$\approx$: Average performance.

[e]✓: Proper performance.

[f]✗: Poor performance.

of transferring between surfaces with different angles. However, the robot is only able to perform 90° convex and concave transition angles; it is not capable of performing higher plane transitions on tight space. TMRs [2,3] have a bigger contact surface, and they are more reliable for overcoming surface irregularities and discontinuities. However, they are not able to negotiate obstacles which are larger than the robot. Robots with step-by-step locomotion show their flexibility in structural climbing. The principle idea is to use multiple grippers interconnected through a chain-like mechanism. They are capable of adapting their bodies to the environment and overcoming obstacles and environmental challenges. Biped wall-climbing robots [4,5] have shown the ability to overcome obstacles. From a design perspective, the former robot is not adaptable in tight spaces, and the latter does not have enough DOFs to travel inside tight and confined spaces with a large number of irregular obstacles. To make small robots faster, brachiation locomotion through step-by-step swinging of the main body is preferred similar to a monkey or a pendulum [6]. The gait of brachiation depends on the initial locomotion speed and the distance between branches, since adequate space for the take-off phase is required. This limitation makes this type of robot not applicable to the intended application. Hybrid robots [7] attempt to take advantage of both continuous and step-by-step movements, such as speed and obstacle overcoming. Existing hybrid robots do not offer greater flexibility in tight spaces. For instance, OmniClimber [8] is a well-developed hybrid robot in which an articulated arm is integrated with an omnidirectional mobile base to move faster on flat surfaces. However, it is merely able to overcome obstacles and wall transitions. The comparison of relevant climbing robots using continuous-motion, step-by-step, and hybrid locomotion can be seen in the Table 1.

Aforementioned robots can hardly meet all the requirements for moving freely inside the train bogies. Nature-inspired step-by-step robots such as [10,11] are more capable to adapt and navigate around obstacles compared with continuous robots. We introduce a multi-modal climbing and manipulation robot, called BogieBot, designed explicitly for locomotion in tight spaces with ferrous metal surfaces, such as a bogie. BogieBot has a redundant chain-link mechanism consisting of six couple joints, four main links, and an inner body link to carry a visual sensor (e.g. a camera or scanner) or a manipulation tool. A preliminary design of the BogieBot mechanism is studied in [12]. This work extends the previous motion analysis and implements a new smart adaptive controller to overcome kinematic barriers due to joint limits and dense surrounding obstacles.

Locomotion in the proximity of obstacles imposes strict requirements in terms of motion control and precise path following. Joint limits and tight surrounding environments introduce internal and external kinematics barriers, which are required to be addressed properly. The goal of control design is to design an adaptive controller for the proposed 12-DOF mechanism, such that it tracks a given desired trajectory precisely. At the same time, the controller needs to maintain the tracking error in predefined safety constraints and satisfy all joint constraints simultaneously. Many of the earlier studies considered state constraints in classical model-based control strategies. The methods mainly prevented constraints transgression by adopting model predictive controls(MPC) [13,14], reference governors [15,16], and the use of set invariance notions [17,18]. Other research has been carried out to design optimal control systems with constant joint position and velocity constraints [19,20]. Apart from recent interest implementing MPC in robotic application, the transition from the process industry to robotics brings challenges since the available computation time is reduced from hours to milliseconds. Computing an optimal control law can become computationally intractable
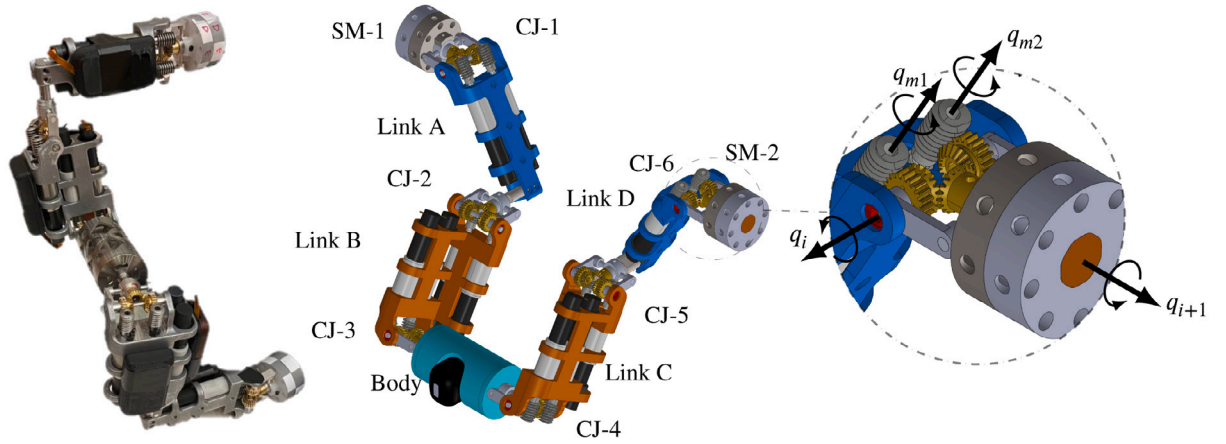
Fig. 1. BogieBot mechanism with six 2-DOF couple joints (CJs). One DOF of CJ-1 & CJ-6 joints are connected to magnets (SM) to mechanically trigger them. One DOF of CJ-3 & CJ-4 joints are used to rotate the Body and sensor.

in large problems because the number of polytopic regions may grow exponentially with the number of constraints. More recently, the Barrier Lyapunov Functions (BLFs) have been employed to handle constraints [21]. Barrier Lyapunov function (BLF) based control approaches have been implemented in general nonlinear single-input single-output (SISO) systems for preventing constraint violation [22]. The extension of BLF-based controllers for multi-input multi-output (MIMO) has also been developed to control manipulator robots [23]. Those controllers guarantee that joint angle and angular velocities remain within their physical limits that are equal to BLF constraints. We are interested, however, in defining additional safety constraints on the joint angles to minimise the violation of BogieBot path following. Time-varying BLF-based (TV-BLF) controllers have also been studied in recent works [24,25] with upper and lower prescribed constraints. The performance of TV-BLF controllers in a few mechanisms with low DOFs, such as a 2-DOF manipulator [26] and a 3-DOF manipulator [25] has been reported. We extended the approach for the proposed redundant climbing robot with six couple joints (12 DOFs) to achieve a safe and collision-free path following.

The main contributions of this paper are threefold. Firstly, we present a novel climbing robot and extended kinematic and motion analysis of the mechanism composed of six 2DoF couple-joints (a totally of 12 DoFs). The proposed unique design allows self-motion of three degree-of-redundancy to achieve secondary behaviours and move close complex obstacles in tight spaces via changing BogieBot's body shape Secondly, we apply such redundancy and self-motion via constructing an adaptive controller with time-varying safety constraints for enhanced trajectory tracking in tight spaces. The TV-BLF is deployed to guarantee that the controller stabilises motion and simultaneously prevents constraints violation. Time-varying safety constraints are considered that begin with large values at the initial conditions (transient period) and quickly converge to small values. Thirdly, the unmodelled dynamic effect of the nonlinear BogieBot is taken into account and estimated using an adaptive radial basis function neural network. Finally, we analyse the robot locomotion and self-motions for specific tasks of wall transitioning and hole crossing inside a Bogie in the rolling stock industry. We show that the controller satisfies joints' constraints and minimises trajectory tracking error for a 12 DOFs mechanism, irrespective of initial conditions and disturbances. The simulation results demonstrate BogieBot's enhanced performance in locomotion and trajectory tracking in tight spaces.
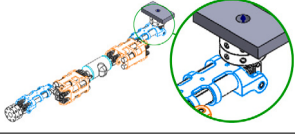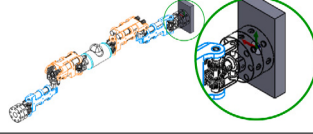
The rest of the paper is organised as follows. Section 2, the mechanical design of the robot is presented as well as its actuation system. Robot kinematics and its self-motion are then analysed in Section 3. Control design and workspace analysis are presented in Sections 4 and 5 respectively, followed by the description and results of simulation experiments which are presented in Section 6. Section 7 concludes the paper.

2. A kinematically redundant climbing mechanism with six couple joints

To climb and explore the three-dimensional confined space of a Bogie, we introduce a miniature multi-modal and kinematically redundant climbing mechanism, called BogieBot, as shown in Fig. 1. The robot's structure consists of six couple joints, four main links, an inner body link (to carry a camera or a tool) and two mechanically switchable magnetic grippers. The design is unique in utilising six couple joints, and its compactness, flexibility and adaptability to move inside tight environments with ferrous surfaces. The size of the robot is 92 mm × 123 mm × 136 mm at off configuration(all joint angles are equal to zero), and it can be extended to 482 mm. BogieBot can perform climbing (or walking) tasks in two steps: chain-link movement and gripper switching. The robot can also rotate or move its main body, which is equipped with a sensor (as shown in Fig. 1), to scan the surroundings from a certain angle or distance.

The 2-DOF couple joint comes with intersecting axes, as illustrated in Fig. 1. In total, the actuation system includes six couple joints which result in twelve independent joints. Each couple joint consists of two brushless DC motors with encoders, two worm and wheel gears, and three bevel gears. The couple joint generates two DOF motions about q_i and q_{i+1} axes (Fig. 1) via speed control

Table 2
Normal and shear forces in the critical cases based on Newton-Euler recursive algorithm.

Normal = 6.8 N Shear = 0.0 N	Normal = 0.0 N Shear = 6.8 N

Main Shaft
Top Holder
16x4mm Magnet
16x4mm Rod
Bottom Holder
Non-slip pad

(a) Top Holder with 8 magnets All configurations.

(b) Bottom Holder (0-degree rotation): ON configuration.

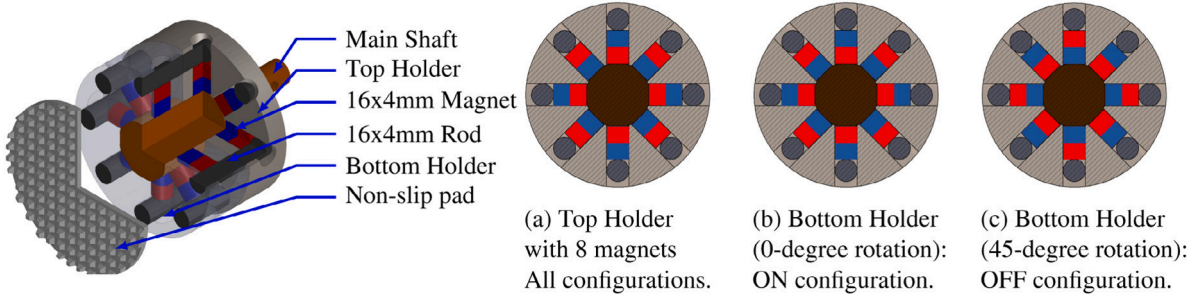(c) Bottom Holder (45-degree rotation): OFF configuration.

Fig. 2. Mechanically switchable magnet gripper; the main shaft drives the bottom holder. Once the bottom holder rotates 45° with respect to the top holder, the direction of magnetic flux changes and cancels through contact surfaces.

and direction of the worms rotation about the q_{m1} and q_{m2} axes. Assuming rigid transmission with no backlash, the relationship between motor and joint velocities is expressed as:

$$\dot{q}_i = k_g(\dot{q}_{m1} + \dot{q}_{m2})/2, \dot{q}_{i+1} = k_g(\dot{q}_{m1} - \dot{q}_{m2})/2 \tag{1}$$

where k_g is the gear ratio (1:20). According to Eq. (1), the output shaft solely revolves around the q_i axis, when two gears are rotated at the same direction and speed. The whole system could also purely revolve around q_{i+1} axis when the worm gears are rotated in the opposite direction but at the same speed. The rotation around q_{i+1} is used to turn on and off the magnet or rotate the attached link.

The couple joint uses the power of both motors under certain circumstances. Each couple joint draws a total of 425.3 mA current for running two DC motors "Faulhaber 1028A012B + 12/4 256:1 + IEM3-1024" each rated at 137.6 mmA and one driver rated at 150 mA. The camera sensor requires 200 mA current. We have considered a total of 6800 mA h power, including two "3.7 V 1000 mA h" and "3.7 V 2400 mA h" batteries, to be placed in links A–D and links B–C, correspondingly. The battery is enough to run six couple joints and sensor for about two hours.

2.1. Switchable magnetic gripper (SMG)

Using magnets as an adhesive technology is common in robots that need to climb surfaces. With this solution, energy consumption must be taken into consideration. We have employed a mechanically switchable magnetic gripper as depicted in Fig. 2, to reduce the power requirements. The proposed gripper governs the ferromagnetic attraction forces by changing the magnetic flux paths. It consists of bottom and top holders. The top holder is fixed, and the bottom holder is driven via the main shaft and rotates relative to the top holder. Each holder consists of eight magnets.

As shown in Fig. 2, magnets are mounted perpendicularly to the shaft on one side and small rods on the other side. The magnets are placed in opposite polarisation directions by one interval. The gripper is activated once the magnets of both holders are parallel with the same polarisation direction. This way, the magnetic fluxes are circulated between rods and the surface, and the attraction force is generated. Once the bottom holder is rotated by 45°, Figs. 2(a) and 2(c), the magnetic fluxes are internally circulated through magnets and rods, and adhesion forces are cancelled. An analysis is done to identify the maximum normal and shear forces at critical robot configurations, as shown in Table 2. To generate the required forces, each gripper consists of 16 magnets with a magnetic force of 0.6 kg each. Considering the lowest coefficient friction of 0.3, each gripper can handle up to 94.17 N normal force and 28.25 N shear forces. These values are significantly more than the critical forces depicted in Table 2.

3. Kinematics and self-motion analysis

Twelve degrees of freedom allow the robot to execute a task with a wide range of possible motions. One DOF of two couple joints, CJ-1 and CJ-6, are connected to magnets to turn them on and off mechanically. One DOF of two couple joints, CJ-3 and CJ-4, is used to rotate the sensor. This allows self-motion of three degree-of-redundancy to achieve secondary behaviour and move over surfaces of complex shapes via changing the BogieBot's body shape.
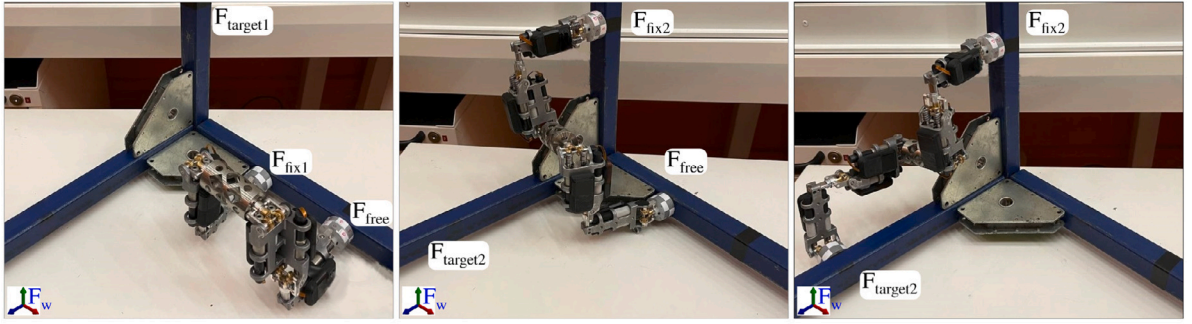
(a) Both ends are attached to the first bar   (b) Move one-end and grip vert. bar   (c) Release other-end and move to 2nd bar

Fig. 3. The sequence of movement from the right horizontal bar to the vertical bar, and then climb to the left horizontal bar.

BogieBot performs climbing via a sequence of motions (Fig. 3). Before and after each motion, both endpoints are fixed to the environment by means of grippers, as shown in Fig. 3(a). Then, one endpoint of the BogieBot is detached from the environment and moved in the three-dimensional space, and then fixed to a different location in the environment (Fig. 3(b)). The motion continues by disconnecting the second endpoint of the BogieBot and moving it to another place (Fig. 3(c)). During each sequence of navigation, two distinguishable configurations can be considered for the robot: open-chain and closed-chain configurations. The BogieBot behaves as an open-chain serial mechanism when only one endpoint is fixed to the environment. A closed chain is formed by attaching both ends to the environment. In this configuration, BogieBot can scan the bogie by moving its embedded sensor along a desired path and angle.

3.1. Forward kinematics

The forward kinematic governs a nonlinear mapping from the joint space to the task space as follows:

$$x(t) = f(q(t)) \tag{2}$$

where x and q are vectors of end-link pose and joint angles, respectively. Velocity of the end-link in task space is a map of robot joints angle by Jacobian matrix:

$$\dot{x}(t) := [v(t), \omega(t)]^\top = J(q)\dot{q}(t) \tag{3}$$

where J is called the geometrical Jacobian matrix of the system and v and ω are linear and angular velocities, respectively. Having forward kinematics, one can use direct differentiation, iterative methods or an explicit form solution to calculate the Jacobian.

The main climbing task is to move the free end of the robot along the desired path. Hence, the direct kinematic is expressed as the pose of BogieBot's free end-link $x(t) = [X \; Y \; Z \; \theta \; \phi \; \psi]^\top \in \mathbb{R}^6$ with respect to its fixed gripper, as a function of joint angles $q(t) \in \mathbb{R}^{12}$. This way, the kinematic equation is represented by a homogeneous transformation matrix, $\mathbf{T}$:

$$^{\text{fixedSMG}}\mathbf{T}_{\text{freeSMG}} = \prod_{j=1}^{j=12} {}^{j-1}\mathbf{T}_j = f(q), \tag{4}$$

where fixedSMG and freeSMG represent fixed and free switchable magnets, j indicates the joint number. $^{j-1}\mathbf{T}_j$, represent the homogeneous transformation matrix between any two adjacent links from frame $\{F\}_j$ to frame $\{F\}_{j-1}$ that is expressed in Denavit–Hartenberg (DH) convention. The DH parameters and their representation can be found in Table 3. The joint positions have been also labelled in Fig. 1.

3.2. Inverse kinematic and self motions of open-chain

Inverse kinematics could be considered as a general damped optimisation problem to reach a continuous and feasible solution in the presence of singular points [27]. This general solution can be obtained by expressing the inverse kinematics problem as:

$$\min_{\dot{q}} \left\{ \|\dot{x} - J\dot{q}\|_{W_a}^2 + \Lambda \|\dot{q} - q_{arb}\|_{W_f}^2 \right\}, \quad \text{subject to } \dot{x} = J\dot{q} \tag{5}$$

where W_a and W_f are positive definite matrices, which are related to the accuracy of tracking and feasibility of tracking in preventing joint velocity limits. Λ is a positive scalar chosen to compromise between feasibility of the joint's velocity and tracking error

respectively. q_{arb} is arbitrary. A range of numerical methods such as LSM, and D-LSM lie in the above definition. For instance by considering $\Lambda = \lambda^2$, $W_a = I$, and $W_f = I$, (5) yield to D-LSM method.

Algorithm 1: Inverse Kinematic

---

1 Initialise: $\dot{x}_d$, q_0 and $\dot{q}_{max}$
2 **for** $t_i \in [t_0, t_e]$ **do**
3   Calculate J and $\sigma_{min}(J)$;
4   **if** $\sigma_{min}(J) > 0$ **then**
5          $J^{(g)} = V\Sigma^\dagger U^T = J^T(JJ^T)^{-1}$
6   **else**
7   **switch** *Solver* **do**
8   **case** *Optimal Λ* **do**
9   Solve $\|\dot{q}^{op}\|^2_{W_f} = \|\dot{q}_{max}\|^2_{W_f}$ to find Λ^{op}
10   **case** *D-LSM* **do**
11   Constant Λ [28]
12         $J^{(g)} = W_f^{-1}J^T\left(JW_f^{-1}J^T + \Lambda W_a^{-1}\right)^{-1}$
13 **Return** $J^{(g)}$

For a redundant system, with $m < n$, and for any predefined $\dot{x}$ there may exist infinite $\dot{q}$ satisfying $\dot{x} = J\dot{q}$. Eventually, a singularity-robust inverse kinematics can be obtained by:

$$\dot{q} = J^{(g)}\dot{x} + \left(I - J^{(g)}J\right)q_{arb}$$
$$J^{(g)} = W_f^{-1}J^T\left(JW_f^{-1}J^T + \Lambda W_a^{-1}\right)^{-1} \tag{6}$$

and the elements of weighted matrix are defined as follows:

$$w_{ii_f} = \begin{cases} 1 + |\frac{\partial H(q)}{\partial q_i}|, & \text{if}\Delta|\frac{\partial H(t)}{\partial q_i}| \geq 0, \\ 1 & \text{if}\Delta|\frac{\partial H(t)}{\partial q_i}| < 0, \end{cases}$$
$$H(q) = \sum_{i=1}^{n}\frac{1}{4}\frac{\left(q_{imax} - q_{imin}\right)^2}{(q_{imax} - q_i)(q_i - q_{imin})} \tag{7}$$

It should be noted that when q_i approaches to its boundary, $\partial H(q)/\partial q_i \to \infty$, consequently the ith element of W_f^{-1} becomes zero and joint number i will be stopped. The first term of (6) is the particular inverse solution of (3) and its second term projects an arbitrary vector q_{arb} to the null space of J, $\mathcal{N}(J)$. This homogeneous solution provides a mathematical tool to handle *self-motions* of BogieBot. This way, the inverse kinematics and the self-motion are solved using Algorithm 1 (see [28]).

4. Adaptive controller with time-varying safety-constraints

The redundant design of BogieBot allows fully connected joint spaces in a tight environment that is desirable for path planning to be found.However, a simple FLC cannot guarantee path following with bounded errors that prevent exceeding physical joint limitations. To clarify this, a small region of the bogie is considered (Fig. 4), to analyse the motion control requirement. RRT* has been employed to find a collision-free path and a FLC is used to track the desired path in simulation. The collision-free space and desired path by RRT* are depicted in Fig. 4, using sample blue points and red lines, respectively. The errors between the actual and desired joint configuration are shown in Fig. 4 It can be seen that there is a valid path (red line) for BogieBot to crawl from the initial configuration to the goal configuration. However, the tracking error is not bounded and varies over time for different joints. The errors jump up to 0.22 degrees at various time instants (grey bars). Unbounded tracking errors could lead to the collision at tight spaces, such as the point highlighted by a blue circle in Fig. 4. FLC demonstrates poor performance close to obstacles and physical joint limits. Given the FLC performance, a more robust control system is essential to guarantee 3D path following with no joint constraints violation. Safety constraints to be employed to avoid two issues. First, the maximum overshoot (usually occur during the transient time) of the output signal (joint angles) must always remain within the physical limits. Second, the tracking error of the output signal (joint angles in here) must remain bounded during the whole path to allow safe navigation close to obstacles. We define time-varying safety constraints as time-varying bounds on the desired trajectory that prescribe physical joint limits and maximum allowable deviations from the desired trajectory.

4.1. Motion control objectives

The primary controller design objective in this work is to allow BogieBot to follow a target trajectory $q_d(t) = [q_{d_1}(t), q_{d_2}(t), \ldots, q_{d_i}(t),$ $\ldots, q_{d_n}(t)]^T$, $n = 12$. At the same time, the controller must guarantee that none of physical joint angle constraints (q_{min}, q_{max}) are

(a) RRT* path planning and FLC based tracking   (b) FLC-based trajectory tracking error(rad)
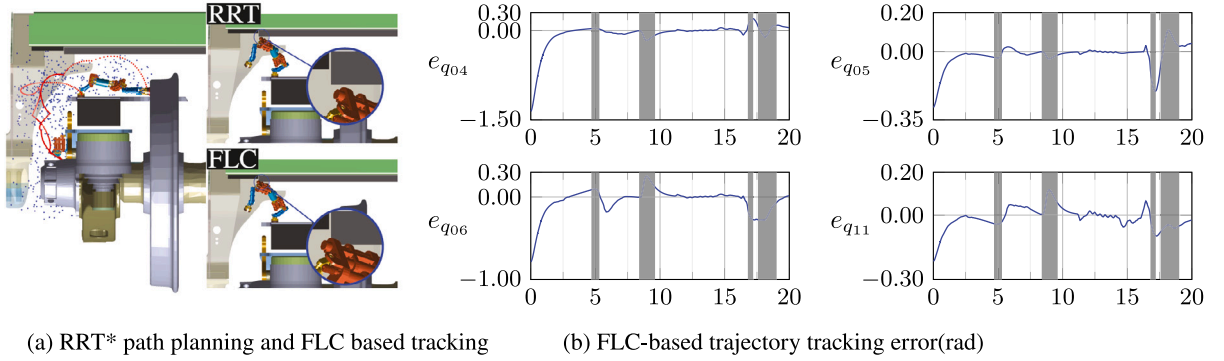
Fig. 4. Poor FLC performance for motion and trajectory tracking in tight spaces. (a) RRT* and FLC are used for path planning and path following at one spot of a bogie. The collision-free path is shown with red points. (b) Tracking errors between the desired and actual joint configurations are depicted in blue. Unbounded tracking errors (highlighted by grey bars) led to violation of physical joint limits and FLC failure close to an obstacle. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)
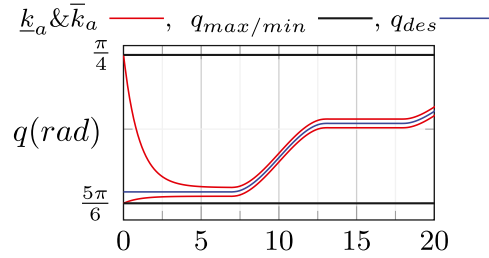

Fig. 5. Time-varying safety-constraints on path following. Consider the desired path (green path) in joint configuration and physical joint limits (black lines).Time varying safety-constraints (red line) are defined by Eq. (9) to bound the tracking errors and prevent violations close to joint limits. In this example, $a_1 = a_2 = a_3 = a_4 = 1$. The final offset ($q_{ss}$) can be set to any small value, e.g. $0.1°$. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

violated and all joint angles always remain bounded in the following region:

$$\Omega_q = \left\{ q(t) \in \mathbb{R}^{n \times 1} \mid \underline{k}_a(t) \leq q \leq \bar{k}_a(t) \right\} \forall t \geq 0 \tag{8}$$

where Ω_q is a bounded set and $\bar{k}_a(t) \leq q_{max}$ and $q_{min} \leq \underline{k}_a(t)$.

To satisfy above inequality and physical joint limits simultaneously, time-varying safety-constraints are defined as:

$$\underline{k}_a(t) = \exp(-a_1 t) q_{min} + (1 - \exp(-a_2 t))(q - q_{ss})$$
$$\bar{k}_a(t) = \exp(-a_3 t) q_{max} + (1 - \exp(-a_4 t))(q + q_{ss}) \tag{9}$$

where a_i and q_{ss} are constant design parameters. The concept of safety-constraints for trajectory tracking is depicted in Fig. 5 for one joint of the robot. Eq (9) offers interesting features to control the output signal quality. A similar but simpler formulation is also employed in [29]. q_{ss} indicates the steady-state error (final error) and the rate of error change is regulated by a_i. Maximum and maximum overshoot is regulated by the value of a_1 and a_3. The settling time is regulated by (a_1 and a_2) and (a_3 and a_4). As the initial configuration is usually far from final configuration, we define a bigger constraint before settling time and smaller constraint after settling time (to reduce the tracking error). Eq. (9) is not a unique solution, and one can employ other functions for defining time-varying barriers with similar behaviour (transition performance and steady-state error).

4.2. Adaptive controller design

The control strategy relies on the time-varying barrier Lyapunov to bound the trajectory errors. It also deploys radial basis function neural network to estimate system parameters of BogieBot in the presence of disturbances and uncertainties. The overall control block diagram is presented in Fig. 6. At a high level, RRT* is used to plan the desired paths in the joint configuration: q_d and $\dot{q}_d$. Initial and goal locations of the contacts are assumed to be known. The proposed recursive TV-BLF based controller with NN estimator consists of three main blocks to be discussed in the following. We used the backstepping approach in two steps to design a virtual controller (α), using Eq. (12), and design the main BLF controller (τ_{act}), using Eq. (13)–(19). Then, Radial Basis Function Neural Network (RBFNN), Eq. (18), is used to approximate and update the unknown function of $U(Z)$, Eq. (14), in real-time, that includes inertia and mass terms of BogieBot dynamic equation motion.
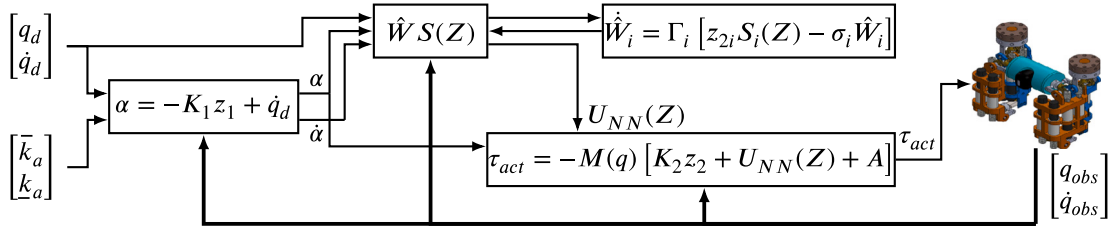
Fig. 6. Block diagram of the closed-loop control system. At the first block, the virtual controller (α) is calculated based on desired and observation states and fed into the NN block. The NN estimates unknown parameters based on the virtual controller and its derivative. In the final block, required joint torques are computed based on the state of feedback, estimated parameters and virtual controller.

The controller design starts with taking into account the dynamic equations of motion, that is expressed as:

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) + \tau_{dis} = \tau_{act} \tag{10}$$

where q, $\dot{q}$, $\ddot{q} \in \mathbb{R}^{n \times 1}$ are the vector of joint angles, velocity, and acceleration, respectively.

$M(q) \in \mathbb{R}^{n \times n}$ is the inertia and mass matrix, $C(q,\dot{q}) \in \mathbb{R}^{n \times n}$ is a matrix of the Coriolis and centrifugal forces, $G(q) \in \mathbb{R}^{n \times 1}$ denotes the vector of gravity forces, $\tau_{act}, \tau_{dis} \in \mathbb{R}^{n \times 1}$ are the joint torque, and disturbance torque, respectively. BogieBot's DH and mechanical parameters are listed in Tables 3 and 4.

Considering $\eta_1 = q$ and $\eta_2 = \dot{q}$, one can represent the dynamics equation in steady space form:

$$\begin{cases} \dot{\eta}_1 & = \eta_2 \\ \dot{\eta}_2 & = M^{-1}[\tau_{act} - C\eta_2 - G - \tau_{dis}] \end{cases} \tag{11}$$

Then, the virtual controller, BLF Controller and NN estimator are designed in three steps as follows:

Step1: Given Eq. (11), we define the joint angle tracking error $z_1 = q - q_d$ and auxiliary error variable $z_2 = \dot{z}_1 = \dot{q} - \alpha$, where α is a virtual controller. Then α is designed to make $z_1 \to 0$, and can be expressed as follows:

$$\alpha = -K_1 z_1 + \dot{q}_d, \tag{12}$$

where $K_1 = \text{diag} \left[k_{11} + \bar{k}_{11}, k_{12} + \bar{k}_{12}, \; \dots \; , k_{1n} + \bar{k}_{1n} \right] \in \mathbb{R}_+^{n \times n}$ and $\bar{k}_{1i}(t) = \sqrt{\left(\dfrac{\dot{k}_{ai}}{k_{ai}} \right)^2 + \left(\dfrac{\dot{k}_{bi}}{k_{bi}} \right)^2 + \beta_i}$.

k_{1i} and β_i are positive design parameters that ensure the time derivatives of α are bounded even when both $\dot{k}_{ai}$ and $\dot{k}_{bi}$ are zero.

Step2: Considering the second term of Eq. (11), the actual controller is designed to stabilise the motions:

$$\tau_{act} = -M(q) \left[K_2 z_2 + U(Z) + A \right] \tag{13}$$

where K_2 is a diagonal tuning parameters.

U is the nonlinear part of BogieBot parameters that is defined as follow:

$$U(Z) = [U_1(Z), \; \dots \; , U_n(Z)]^\top = -M(q)^{-1}[\tau_{dis} + C(q,\dot{q})\dot{q} + G(q) + M(q)\dot{\alpha}] \quad , \tag{14}$$

where

$$\mu_i = \frac{h_{1i}}{k_{bi}^{2r} - z_{1i}^{2r}} + \frac{1 - h_{1i}}{k_{ai}^{2r} - z_{1i}^{2r}} \quad , \quad h(z_{1i}) := \begin{cases} 1 & z_{1i} > 0 \\ 0. & z_{1i} \leq 0, \end{cases} \quad \text{and } A = [\mu_1 z_{11}^{2r-1}, \mu_2 z_{12}^{2r-1}, \; \dots \; , \mu_n z_{1n}^{2r-1}]^T \tag{15}$$

Step3: Next the unknown function $U(Z)$, $Z = [q^T, \; \dot{q}^T, \; \alpha^T, \; \dot{\alpha}^T]^T$, is estimated utilising a RBFNN based method. The unknown function $U(Z)$ has been modelled as:

$$U_i(Z) = W_i^{*T} S_i(Z) + \varepsilon_i(Z) \tag{16}$$

where $S_i(Z) = [S_{i1}(Z), S_{i2}(Z), \; \dots \; , S_{is_i}(Z)]^\top \in \mathbb{R}^{s_i}$, $s_i > 1$ is the number of nodes in the RBFNN.

$S_{ij}(Z)$ is selected as a Gaussian function which is given by

$$S_{ij}(Z) = \exp \left(-(Z - Z_{ij})^\top (Z - Z_{ij}) / \varphi_i^2 \right), \tag{17}$$

where Z_i is the centre of the ith input and φ_i is the width of the Gaussian function. $\varepsilon_i(Z)$ is the neural network approximation error, whereas W_i^* is the optimal neural network weight for which $\varepsilon_i(Z)$ would be the smallest. Since $U(Z)$ is unknown, W^* is also unknown. Therefore, the RBFNN has to be implemented as

$$U_{i,NN} = \hat{W}_i^\top S_i(Z), \tag{18}$$

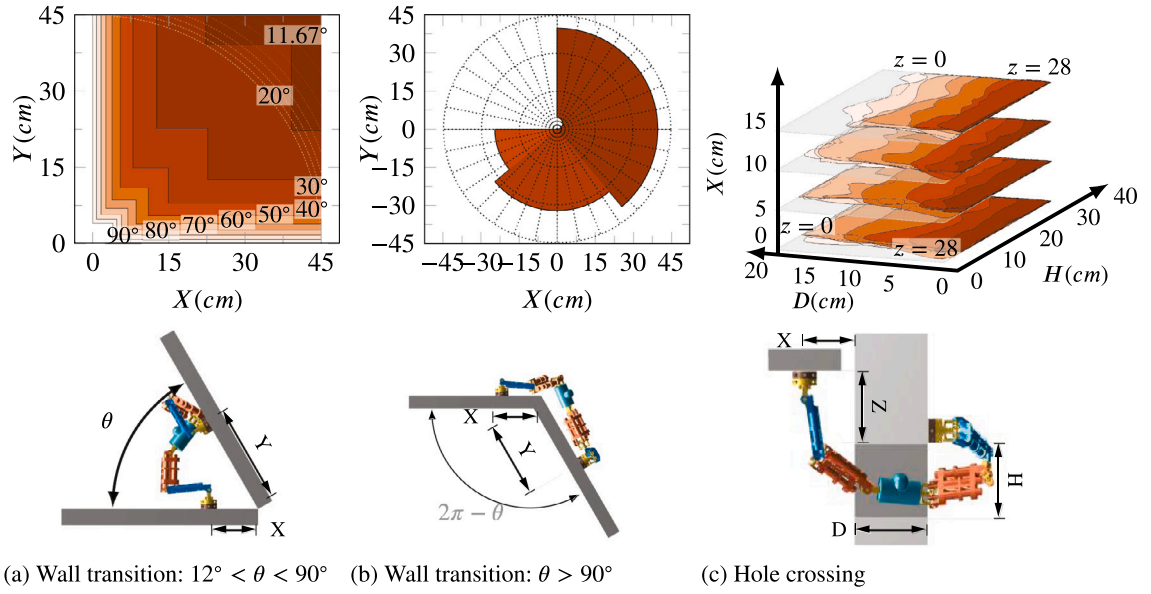(a) Wall transition: $12° < \theta < 90°$ (b) Wall transition: $\theta > 90°$ (c) Hole crossing

Fig. 7. Wall transition and hole crossing analysis of the proposed climbing robot with six 2-DOF couple joints.

where $\hat{W}$ is an estimate for W^*. To make sure that estimation of W^* always generates stabiliser actuator torques, the adaptive law can be used to update ith [24]:

$$\dot{\hat{W}}_i = \Gamma_i \left[z_{2i} S_i(Z) - \sigma_i \hat{W}_i \right] \qquad (19)$$

where $\Gamma_i = \Gamma_i^\top > 0$ is an arbitrary positive matrix, and $\sigma_i > 0, i = 1, \ldots, n$ is an arbitrary small and positive constant.

To guarantee that the controller stabilises the motion and prevent constraints violation, the following TV-BLF has been considered:

$$V_1(z_1) = \frac{1}{2r} \sum_{i=1}^{n} \log \frac{1}{1 - \xi_i^{2r}} + \frac{1}{2} z_2^\top z_2 + \frac{1}{2} \sum_{i=1}^{n} \tilde{W}_i^T \Gamma_i^{-1} \tilde{W}_i. \qquad (20)$$

The first and second terms are selected to guarantee the stability of virtual controller, (Eq. (12)), and actual controllers, (Eq. (13)), respectively. The last term is used to stabilise the Neural Network based part of adaptive law Eq. (19). Stability analysis for n-DOF manipulators and implementation for a two-DOF manipulator can be found in [24].

5. Workspace analysis and obstacle negotiation

The proposed mechanism offers a remarkable ability to move between walls. In this section, we further investigate its surface transition ability based on workspace analysis in three situations; concave and convex wall transitions and hole crossing. For this analysis, the maximum length of the wall is considered to be equal to the maximum length of BogieBot, i.e. 48.2 cm. For concave walls with different angles (θ), the reachable distances on the target wall (Y) with respect to the base offset (X) have been depicted in Fig. 7(a). The minimum passable wall angle is 11.67°, if the robot is placed far enough from the intersection of the walls ($X > 220$ mm). The reachable locations on the target wall are increased by enlarging the walls' angle. Almost everywhere is reachable for perpendicular walls. In the convex transition, the free end must always be parallel with the target wall (Fig. 7(b)). The maximum reachable distance in Y is 45.4 cm when the wall angle is 90° or 230°. By increasing the wall's angle, the boundary of reachable space will decrease slightly to prevent collision with the base wall. Maximum distance decreases to 32.6 cm for angles between 230° and 320°, and to 25 cm for angles between 230° and 320°. Hole crossing in 3D space is still a challenging task that depends on hole size, robot location and target point. In this analysis, we run a Monte Carlo simulation with various locations of base ($X < 20$ cm), offset ($Z < 35$ cm), height ($H > 5.6$ cm) and depth ($D < 22$ cm) of the hole. The simulation result for a hole with constant width has been depicted in Fig. 7(c). As can be seen from Fig. 7(c), reachable regions strongly depend on the Z offset. For example, at $X = 0$ and $Z = 0$, the reachable region is almost independent of the height and depth of the hole. However, the region shrinks at $Z = 0$ (maroon region). Similar behaviour can be found by increasing the X offset.
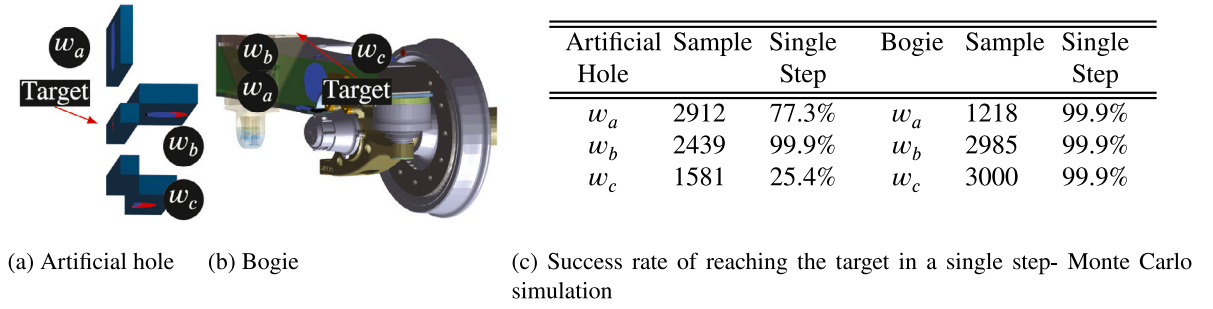
Artificial Hole	Sample	Single Step	Bogie	Sample	Single Step
w_a	2912	77.3%	w_a	1218	99.9%
w_b	2439	99.9%	w_b	2985	99.9%
w_c	1581	25.4%	w_c	3000	99.9%

(a) Artificial hole   (b) Bogie   (c) Success rate of reaching the target in a single step- Monte Carlo simulation

Fig. 8. Adaptiveness to various conditions and initial positions (w_a, w_b, w_c). (a) Simulation trials via consideration of an artificial hole to find the minimum passable hole configuration by BogieBot. (b) Simulation trials using CAD model of a tightest space on the bogie. The robot to climb from random points on w_a to w_b and then move to w_c. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

6. Simulation experiments and results

To validate the designed mechanism with the proposed adaptive control approach, Fig. 6, various simulations using MATLAB and Simulink are conducted. Simulations are done via consideration of two virtual environments, one is with artificial structures and holes and the other one is with an exact CAD model of a bogie. As a necessary simplification for the initial study, we employed offline path planning methods with the help of the CAD model, assuming inside the Bogie is static.

6.1. Robot adaptiveness for structural climbing in various conditions

This experiment evaluates the mechanism's adaptiveness and self-motion to climb in various conditions and squeeze through narrow passages. In the first scenario, the robot's task is to cross through a hole modelled by 4 horizontal blocks, as shown in Fig. 8.(a). The second locomotion experiment is conducted around a hard to reach area of the bogie, which is depicted in Fig. 8.(b). Employing Monte Carlo simulation, trials were made with different starting locations and configurations, to take into account the uncertainties. The initial positions were distributed on three different horizontal and vertical surfaces, w_a, w_b and w_c, respectively. For each surface, 3000 initial positions are uniformly sampled within a circle of 10 cm, and the simulation was performed for all feasible samples. Samples are discarded if they are not lying on the environment's surface or if there is a collision between the robot and the environment at the initial configuration. For example, the number of valid points on the w_a region of the bogie is 1218, as the surface on the bogie is so tiny. The number of samples and results for each case are summarised in Fig. 8(c) and discussed below. For the majority of feasible initial positions on bogie, 99,9%, and relatively high rate of samples on artificial environment, 77.3% (w_a), 99,9% (w_b), and 25,4% (w_c), the robot reached the target in one single step with position error less than 1 mm (blue samples). For the remainder of the samples, the robot can reach the target in a single step with a position error larger than 1 mm or multi-steps with position error less than 1 mm (red samples).

6.2. Path following response to safety-constraints and various initial configurations

Path following in the presence of initial errors with respect to desired joint configurations is critical, particularly if the system consists of many joints. This experiment investigates the effectiveness of the proposed controller for path following, with given desired safety-constraints and physical joint constraints of the BogieBot. The robot controller is commanded to track seven trajectories, each initialised from different robot positions to a single desired target. The physical constraints are different among multiple joints of BogieBot, and the offset value of safety constraints is set to 0.1 degrees. The results in Fig. 9 show that BogieBot can be properly controlled from random configuration to goal configuration. As can be seen from Fig. 9, joint angles always remain in the prescribed constraints, irrespective of different initial configurations. Moreover, the robot smoothly tracked the path close to physical limits but never violated the limits, see q_{05} from 0–20 s, q_{08} from 12–20 s and q_{08} from 0–8 s. It should be noted that the controller is implemented without external disturbance. Hence, the actuator torques are very smooth (Fig. 10)

6.3. Controller performance in the presence of disturbance and unmodelled dynamic effects

This experiment examines controller robustness in the presence of inaccuracy and unmodelled effects in BogieBot's dynamics. Specifically, the controller is tested by adding external disturbance (1% unknown torque), and 10% parametric inaccuracies (in C and G matrices of Eq. (10)). This experiment is conducted around the hardest-to-reach spots of the bogie, and the result of the proposed controller is depicted in Fig. 11. The robot's task is to move up its end from part A to part B and then move the other end and attach it to part C. It should be emphasised that the neural network has four inputs and each input has 12 dimensions.
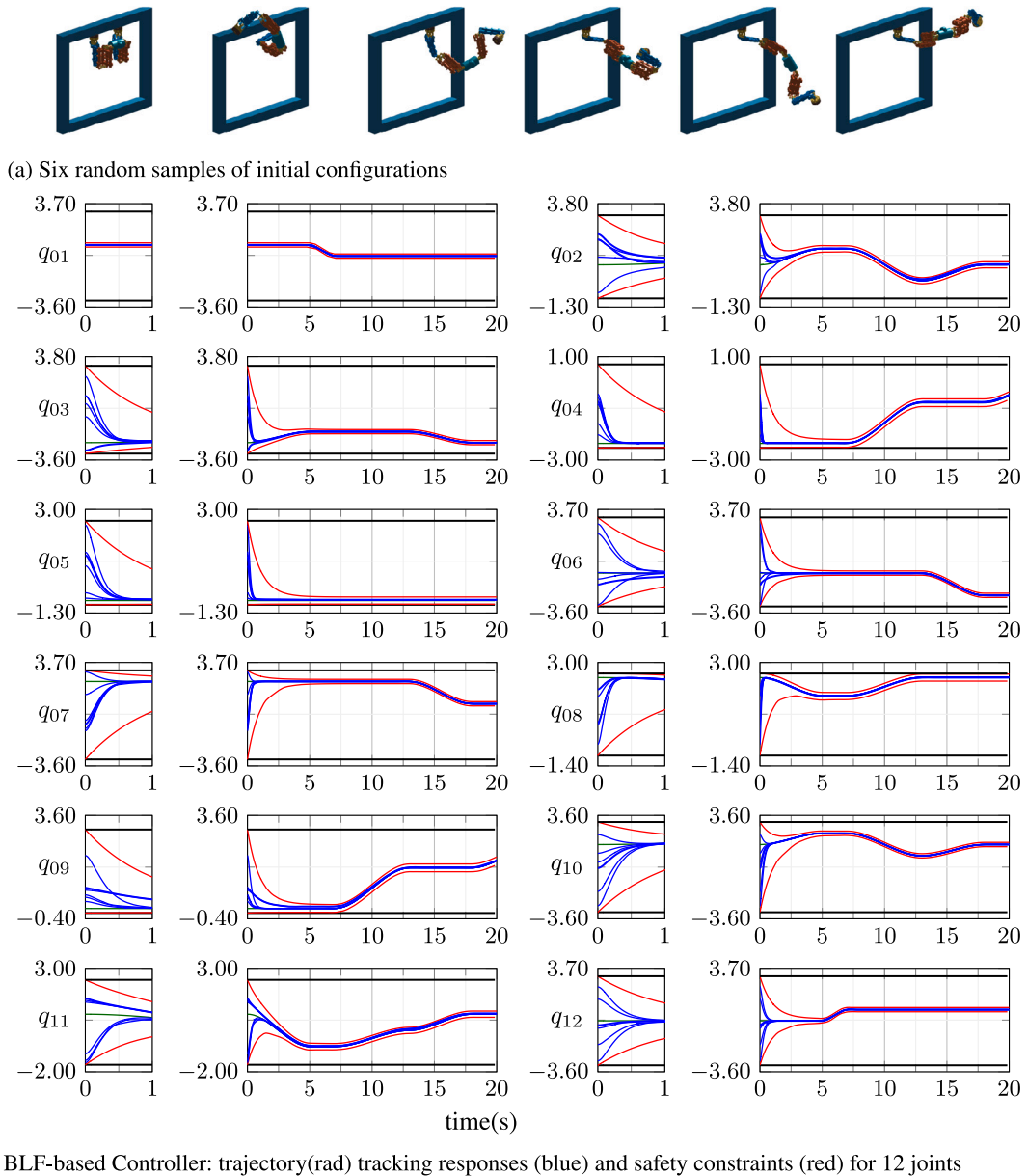
(a) Six random samples of initial configurations

(b) BLF-based Controller: trajectory(rad) tracking responses (blue) and safety constraints (red) for 12 joints

Fig. 9. Path following responses of the proposed controller (b) to various initial configurations (a). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

The neural network architecture has 3 layers with 5 nodes per dimension (240 nodes in total). The centres are chosen in the area of joint limits for the first twelve dimensions $[-5, 5]$ for the next twelve dimensions and between $[-1, 1]$ for last 24 dimensions. Variance of centres are set as $\varphi^2 = 9$. The initial weights $\hat{W}_i, i = 1, 2, \ldots, 240$ are chosen randomly. The control parameters are chosen as $K_1 = 400I_{12\times12}$, $K_2 = 900I_{12\times12}$, $\sigma_i = 0.002$, $\Gamma_i = 800I_{240\times240}$, which satisfy the conditions. The control scheme requires no preliminary offline learning phase. Weight training occurs online. RBF neural network consists of a nonlinear mapping from input to output, but a linear mapping from the hidden layer space to the output space that can significantly speed up the learning process to approximate any continuous function and avoid local minima. The tuning algorithm is an unsupervised backpropagation. The stability of the controller has been proved based on Lyapunov theory regardless of the value of NN weights, which are initialised at zero or randomly. The controller holds the system stable until the NN begins to learn. The NN weights are tuned online in real-time as the "stabilized" system tracks the desired trajectory; as the NN learns U, the tracking performance improves. Aside from the commencement of motion, the results show a very small and bounded error between the achieved joint angles and the desired ones. The performance of proposed BLF-based method is depicted in Fig. 12. Furthermore, the tracking errors of proposed
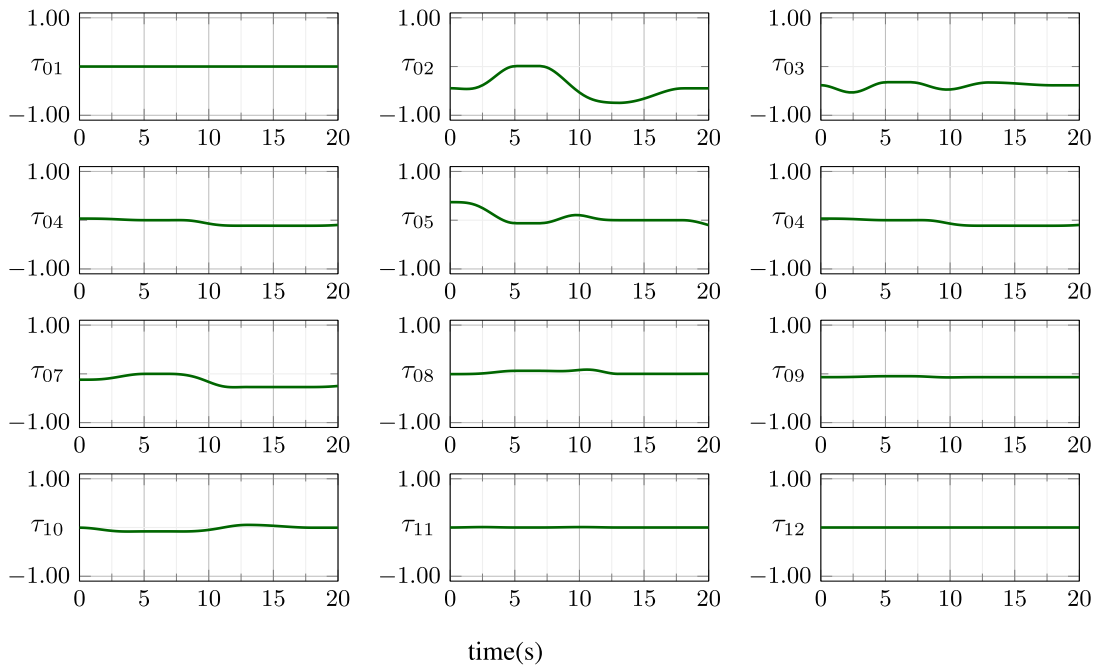
Fig. 10. Joint torques (Nm) based on the TV-BLF controller in the presence of no external load disturbances.
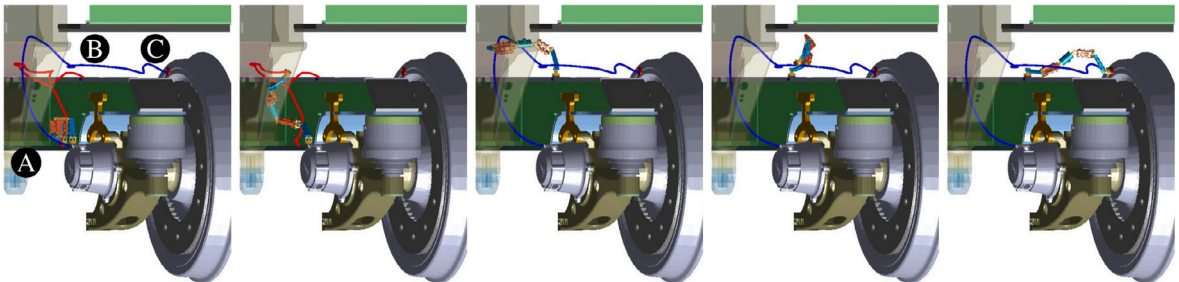

Fig. 11. Sequences of a two-step simulation trial in the bogie. First step: red path, Second step: blue path.

BLF-based method methods and Quadratic-Lyapunov-Function-based controller are indicated in Fig. 13. The comparison of tracking errors clearly shows the superior performance of the proposed BLF-based controller to keep the errors within the safety-constraints (0.1 degrees). QLF errors are varying over time and jump to 0.3 degrees at some time instances which could lead to unexpected robot collision with the environment. Finally, the actuators' torques in the presence of disturbances are demonstrated in Fig. 14. The torques are feasible but they are not smooth. This performance is expected as the controller compensates for the disturbances.

7. Conclusion and future works

Current practices for inspection of undercarriage vehicles are mostly manual and time-based maintenance (TBM). A miniature robot, called BogieBot, is proposed for automated inspection. The robot can climb inside a bogie and manipulate the position of its sensor for data acquisition. The robot mechanism design is unique in utilising six couple joints, and its flexibility and adaptability in performing locomotion inside tight spaces with ferrous surfaces. Aside from robot size and its uniqueness in mobility, we controlled the robot locomotion and self-motions by constructing an adaptive neural network TV-BLF based controller. The method enhances trajectory tracking by bounding and suppressing the propagation of tracking errors. Extended analyses on the forward and inverse kinematic, work-space, and self-motion of BogieBot are presented. The analyses and simulations show robot mobility and flexibility in tight 3D spaces, for overcoming obstacles and environmental challenges (such as surface transition, irregularities, and discontinuities). Since BogieBot navigates close to obstacles, reliable motion planning and precise path following are critical
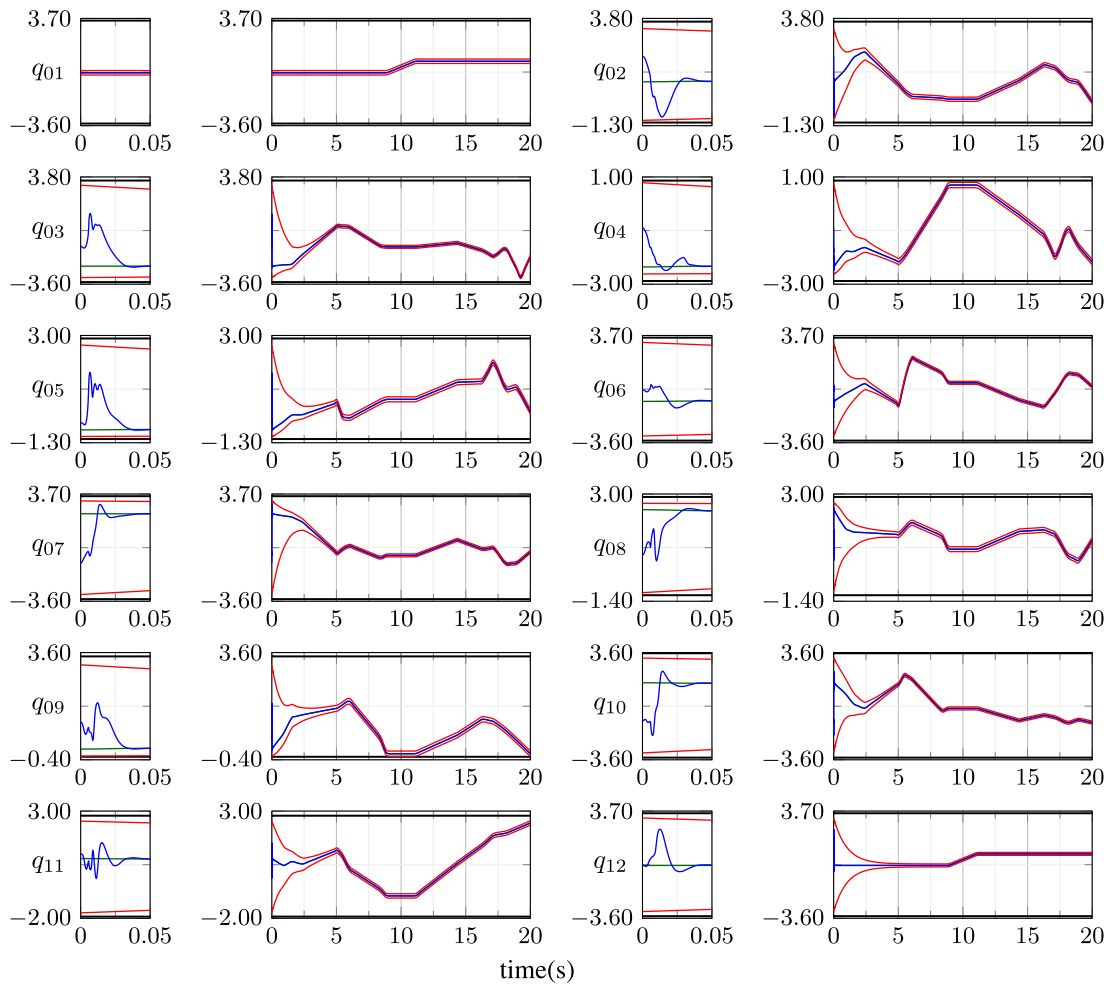
Fig. 12. Trajectory (rad) tracking: proposed BLF-based Controller (green) versus QLF-based Controller (blue) in the presence of disturbances. Time-varying BLF-based safety constraints are shown in red. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

for collision avoidance. The proposed control scheme enhances robot performance in collision-free manoeuvring, by minimising trajectory tracking error irrespective of the initial conditions. The results also show controller robustness in rejecting the external disturbances and estimating unknown parameters online.

Future work will include testing the robot prototype in actual field settings and deploying it inside a bogie. We can further enhance the mechanism design by adding variable stiffness mechanisms to its couple joints and grippers. For instance, to design flexible couple joints with mechanical spring coupling between the motor and the actuator and to allow the stiffness change for the spring coupling. For real-time navigation, the offline multi-step global path planning (footstep planning) can be accomplished by an online replanning of single-step paths between two footsteps. Online replanning of single paths can compensate for position errors and reliable obstacle avoidance at a low computation cost. Additional miniature sensors such as embedded event sensors can be employed for navigation and mapping under low light conditions. Reinforcement learning can be also applied for improving motion planning, controller optimisation, and scenario-based learning policies inside a Bogie.
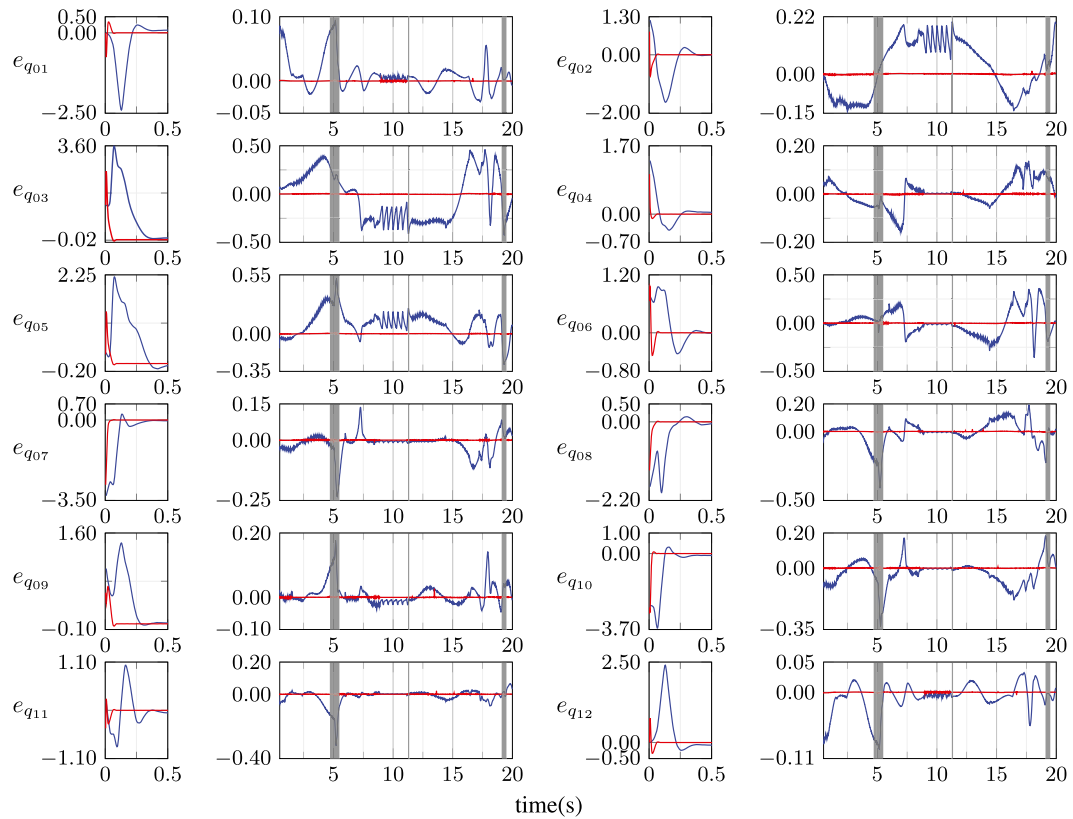
Acknowledgements

Fig. 13. Tracking-error (rad) comparison between BLF based controller (red) and QLF based controller (blue). Unbounded tracking errors (grey bars) led to violation of physical joint limits and QLC failure close to an obstacle. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)
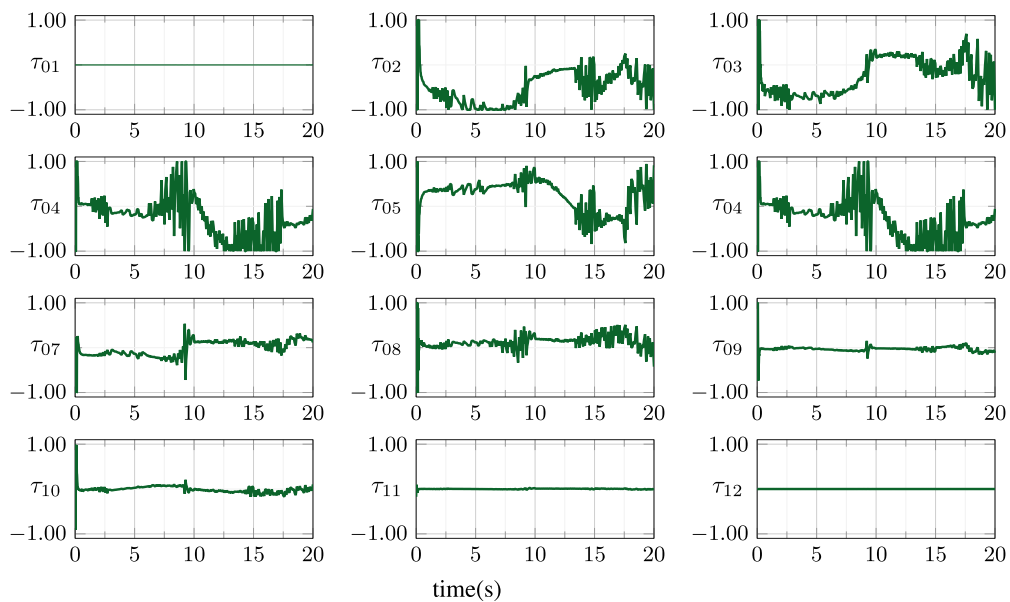

Fig. 14. Joint torques(Nm) based on BLF based controller in the presence of external load (disturbances).

Appendix

Table 3
DH parameters.

Link	a_i	α_i	d_i	q_i	q_{lim}	Link	a_i	α_i	d_i	q_i	q_{lim}
1	0	0°	10	q_{11}	−180:180	2	0	−90°	28.5	q_{12}	−50:185
3	79.46	0°	0	q_{21}	−180:180	4	0	90°	37	q_{22}	−170:40
5	76	−90°	2	q_{31}	−55:145	6	0	90°	−18	q_{32}	−180:180
7	0	0°	47.5	q_{41}	−180:180	8	0	90°	47.5	q_{42}	−55:145
9	76	−90°	18	q_{51}	−10:170	10	0	90°	12	q_{52}	−180:180
11	79.46	0°	37	q_{61}	−90:140	12	0	90°	28.5	q_{62}	−180:180

Table 4
Mechanical properties.

Part	m (g)	Ix (g mm^2)	Iy (g mm^2)	Iz (g mm^2)
Link A&D	31.19	33.8e−2	194.3e−2	209.2e−2
Link B&C	52.22	161.7e−2	293.6e−2	380.6e−2
Couple joints	11.46	5.3e−2	7e−2	10.9e−2
Body	121.55	210.9e−2	406e−2	433e-2

The homogeneous transformation matrix between two adjacent links from frame $\{F\}_j$ to frame $\{F\}_{j-1}$ is expressed as

$$
{}^{j-1}\mathbf{T}_j = \begin{bmatrix} c_{\psi_j} & -s_{\psi_j} c_{\alpha_j} & s_{\psi_j} s_{\alpha_j} & a_j c_{\alpha_j} \\ s_{\psi_j} & c_{\psi_j} c_{\alpha_j} & -c_{\psi_j} s_{\alpha_j} & a_j s_{\alpha_j} \\ 0 & s_{\alpha_j} & c_{\alpha_j} & d_j \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad \psi_j = q_j + q_j^{\text{off}}
\tag{21}
$$

where c_{ψ_j} and s_{ψ_j} indicate $\cos(\psi_j)$ and $\sin(\psi_j)$ respectively, and q_j, d_j, a_j, and α_j are the rotation along z, translation along z, translation along x, rotation along x. q_j^{off} is also rotation along z when robot is off.

References

[1] Fabien Tâche, F. Pomerleau, Gilles Caprari, Roland Y. Siegwart, Michael Bosse, Roland Moser, Three-dimensional localization for the MagneBike inspection robot, J. Field Robot. 28 (2011) 180–203.

[2] Keng Huat Koh, M. Sreekumar, S.G. Ponnambalam, Hybrid electrostatic and elastomer adhesion mechanism for wall climbing robot, Mechatronics 35 (2016) 122–135.

[3] Haocai Huang, Danhua Li, Zhao Xue, XianLei Chen, Shuyu Liu, Jianxing Leng, Yan Wei, Design and performance analysis of a tracked wall-climbing robot for ship inspection in shipbuilding, Ocean Eng. 131 (2017) 224–230.

[4] Mark Minor, Hans Dulimarta, Girish Danghi, Ranjan Mukherjee, R Lal Tummala, Dean Aslam, Design, implementation, and evaluation of an under-actuated miniature biped climbing robot, in: Proceedings. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2000)(Cat. No. 00CH37113), Vol. 3, IEEE, 2000, pp. 1999–2005.

[5] Yisheng Guan, Haifei Zhu, Wenqiang Wu, Xuefeng Zhou, Li Jiang, Chuanwu Cai, Lianmeng Zhang, Hong Zhang, A modular biped wall-climbing robot with high mobility and manipulating function, IEEE/ASME Trans. Mechatronics 18 (6) (2012) 1787–1798.

[6] Chi-Ying Lin, Zong-Han Yang, TRBR: Flight body posture compensation for transverse ricochetal brachiation robot, Mechatronics 65 (2020) 102307.

[7] Giuk Lee, Kunchan Seo, Seokwoo Lee, Junhwan Park, Hwang Kim, Jongwon Kim, TaeWon Seo, Compliant track-wheeled climbing robot with transitioning ability and high-payload capacity, in: IEEE International Conference on Robotics and Biomimetics, IEEE, 2011, pp. 2020–2024.

[8] Mahmoud Tavakoli, Joao Lourenco, Carlos Viegas, Pedro Neto, Anibal T de Almeida, The hybrid OmniClimber robot: wheel based climbing, arm based plane transition, and switchable magnet adhesion, Mechatronics 36 (2016) 136–146.

[9] Fumin Gao, JianChun Fan, Laibin Zhang, Jiankang Jiang, Shoujie He, Magnetic crawler climbing detection robot basing on metal magnetic memory testing technology, Robot. Auton. Syst. 125 (2020) 103439.

[10] Tin Lun Lam, Yangsheng Xu, A flexible tree climbing robot: Treebot-design and implementation, in: IEEE International Conference on Robotics and Automation, IEEE, 2011, pp. 5849–5854.

[11] Il Hwan Han, Hoon Yi, Chang-Woo Song, Hoon Eui Jeong, Seung-Yop Lee, A miniaturized wall-climbing segment robot inspired by caterpillar locomotion, Bioinspiration Biomim. 12 (4) (2017) 046003.

[12] Mohammad Adinehvand, Ehsan Asadi, Chow Y. Lai, Hamid Khayyam, Kevin Tan, Reza Hoseinnezhad, BogieBot: A climbing robot in cluttered confined space of bogies with ferrous metal surfaces, in: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), IEEE, 2021, page Inproceedings.

[13] Frank Allgöwer, Alex Zheng, Nonlinear Model Predictive Control, Vol. 26, Birkhäuser, 2012.

[14] David Q Mayne, James B Rawlings, Christopher V Rao, Pierre OM Scokaert, Constrained model predictive control: Stability and optimality, Automatica 36 (6) (2000) 789–814.

[15] Alberto Bemporad, Reference governor for constrained nonlinear systems, IEEE Trans. Automat. Control 43 (3) (1998) 415–419.

[16] Elmer Gilbert, Ilya Kolmanovsky, Nonlinear tracking control in the presence of state and control constraints: a generalized reference governor, Automatica 38 (12) (2002) 2063–2073.

[17] Tingshu Hu, Zongli Lin, Control Systems with Actuator Saturation: Analysis and Design, Springer Science & Business Media, 2001.

[18] Derong Liu, Anthony N. Michel, Dynamical Systems with Saturation Nonlinearities: Analysis and Design, Springer-Verlag, 1994.

[19] Bidyadhar Subudhi, Santanu Kumar Pradhan, Direct adaptive control of a flexible robot using reinforcement learning, in: 2010 International Conference on Industrial Electronics, Control and Robotics, IEEE, 2010, pp. 129–136.
[20] Yunong Zhang, Shuzhi Sam Ge, Tong Heng Lee, A unified quadratic-programming-based dynamical system approach to joint torque optimization of physically constrained redundant manipulators, IEEE Trans. Syst. Man Cybern. B 34 (5) (2004) 2126–2132.
[21] Khoi B. Ngo, Robert Mahony, Zhong-Ping Jiang, Integrator backstepping using barrier functions for systems with multiple state constraints, in: Proceedings of the 44th IEEE Conference on Decision and Control, IEEE, 2005, pp. 8306–8312.
[22] Keng Peng Tee, Shuzhi Sam Ge, Eng Hock Tay, Barrier Lyapunov functions for the control of output-constrained nonlinear systems, Automatica 45 (4) (2009) 918–927.
[23] Wei He, Yuhao Chen, Zhao Yin, Adaptive neural network control of an uncertain robot with full-state constraints, IEEE Trans. Cybern. 46 (3) (2015) 620–629.
[24] Yan-Jun Liu, Shumin Lu, Shaocheng Tong, Neural network controller design for an uncertain robot with time-varying output constraint, IEEE Trans. Syst. Man Cybern. Syst. 47 (8) (2016) 2060–2068.
[25] Zhiqiang Ma, Panfeng Huang, Adaptive neural-network controller for an uncertain rigid manipulator with input saturation and full-order state constraint, IEEE Trans. Cybern. (2020).
[26] Yuxiang Wu, Rui Huang, Xian Li, Song Liu, Adaptive neural network control of uncertain robotic manipulators with external disturbance and time-varying output constraints, Neurocomputing 323 (2019) 108–116.
[27] Yoshihiko Nakamura, Hideo Hanafusa, INverse Kinematic Solutions with Singularity Robustness for Robot Manipulator Control, IEEE, 1986.
[28] Tomomichi Sugihara, Solvability-unconcerned inverse kinematics by the Levenberg–Marquardt method, IEEE Trans. Robot. 27 (5) (2011) 984–991.
[29] Yiannis Karayiannidis, Zoe Doulgeri, Model-free robot joint position regulation and tracking with prescribed performance guarantees, Robot. Auton. Syst. 60 (2) (2012) 214–226.