Robotica (2017) volume 35, pp. 463–482. © Cambridge University Press 2015
doi:10.1017/S0263574715000685

An optimal wheel-torque control on a compliant modular robot for wheel-slip minimization

Avinash Siravuru*, Suril V. Shah and K. Madhava Krishna

Robotics Research Centre, International Institute of Information Technology, Hyderabad, 500032, India.
E-mails: avinash.siravuru@research.iiit.ac.in, surilshah@iiit.ac.in, mkrishna@iiit.ac.in

(Accepted July 20, 2015. First published online: September 1, 2015)

SUMMARY
This paper discusses the development of an optimal wheel-torque controller for a compliant modular robot. The wheel actuators are the only actively controllable elements in this robot. For this type of robots, wheel-slip could offer a lot of hindrance while traversing on uneven terrains. Therefore, an effective wheel-torque controller is desired that will also improve the wheel-odometry and minimize power consumption. In this work, an optimal wheel-torque controller is proposed that minimizes the traction-to-normal force ratios of all the wheels at every instant of its motion. This ensures that, at every wheel, the least traction force per unit normal force is applied to maintain static stability and desired wheel speed. The lower this is, in comparison to the actual friction coefficient of the wheel-ground interface, the more margin of slip-free motion the robot can have. This formalism best exploits the redundancy offered by a modularly designed robot. This is the key novelty of this work. Extensive numerical and experimental studies were carried out to validate this controller. The robot was tested on four different surfaces and we report an overall average slip reduction of 44% and mean wheel-torque reduction by 16%.

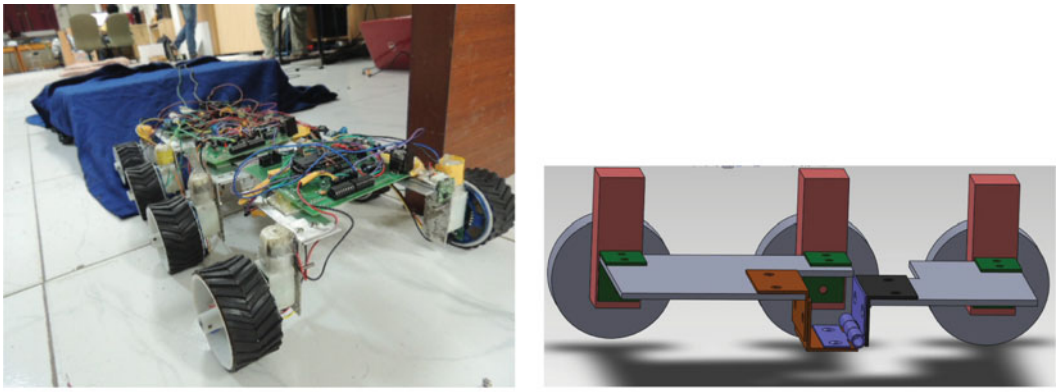
KEYWORDS: Modular Robots, Wheel-Slip Minimization, Compliant Robot

Nomenclature

ϕ_i	Relative angle between links i and $i+1$
τ_{wi}	Torque applied at wheel i
θ_i	Absolute angle of link i
c	Clearance between wheel center and trunk
F_i	Traction force at Wheel i
fx_{li}	X component of the trunk-joint reaction force between trunks i and $i+1$
fx_{wi}	X component of the wheel-joint reaction force on the trunk due to wheel i
fy_{li}	Y component of the trunk-joint reaction force between trunks i and $i+1$
fy_{wi}	Y component of the wheel-joint reaction force on the trunk due to wheel i
k_i	Spring constant of spring i
l	Length of the trunk
l_0	Offset distance between trunk-joint and wheel joint
m_l	Mass of the trunk
m_w	Mass of the wheel
N_i	Normal force at Wheel i
r	Radius of the wheel
S_i	Spring i
W_i	Wheel i
w_l	Weight of the trunk ($m_l g$)
w_w	Weight of the wheel-pair ($2m_w g$)

* Corresponding author. E-mail: sirneonash@gmail.com

https://doi.org/10.1017/S0263574715000685 Published online by Cambridge University Press



(a) (b)

Fig. 1. Snapshots of the (a) compliant modular robot and (b) the sectional view of a 2-module CAD model.

1. Introduction

Wheeled robots with enhanced traversing ability will play a crucial role in search and rescue operations, planetary exploration, etc. It is highly desired that the robot sustains forward motion without losing stability throughout its operation. The extent to which a robot can achieve this on a general unstructured terrain defines its Terrainability.[1] In a search and rescue operation, the terrain knowledge is generally unknown beforehand. Therefore, it is important and useful to have high terrainability. We have proposed a novel compliant modular robot that can ascend obstacles up to three times its wheel diameter. The robot's design details and its climbing ability were shown in ref. [2]. We later expanded this compliant joint design to enable the robot to successfully descend from equally big-sized obstacles. However, it was observed that the conventional velocity controllers used for generating forward motion resulted in considerable amount of wheel-slip while climbing steep obstacles (like steps). Wheel-slip typically hinders continuous forward motion and severely affects the accuracy of wheel odometry. Therefore, this work attempts to improve the terrainability of the robot by minimizing wheel-slip. A viable solution to this problem is to proactively maintain the least possible traction-to-normal force ratios at all the wheels throughout the traversal. This objective was formulated as a wheel-torque optimization problem and the underlying theory was presented in ref. [3]. Multiple objective functions that capture the notion of wheel-torque minimization were considered and their respective merits and demerits were presented. In the present work, we build a controller based on the wheel-torque optimizer developed in ref. [3]. and apply it to the numerical and experimental models of the robot to validate its efficacy.

Wheel-slip minimization is well studied for SHRIMP[4,5] and CRAB[6–9] robots. However, it may be noted that the direct application of the formalism shown in these works to a compliant modular robot is non-trivial. In the case of the above-mentioned single module robots, the wheels always maintain contact with the ground during rough terrain traversal. Therefore, a generalized set of static stability equations can be derived to analyze the robot's motion. However, in the case of multi-module compliant robot, wheel-ground contact is not always guaranteed or desirable.[2] Accordingly, appropriate changes take place in the static stability equations. This is systematically analyzed in this paper. Additionally, extensive numerical and experimental studies were conducted, and a consistent and significant reduction in slip rate was noted after the application of the new optimal wheel-torque based controller. In contrast to the results shown in the previous work,[8] the slip rate reduction is consistently significant along a wide range of coefficients of friction. This further validates the advantages of a modular and compliant design. This forms the main contribution of this work. Also, a notable reduction in mean torque requirement was noted at all the wheels, thereby improving the energy efficiency of the robot. Figure 1 shows the prototype of the proposed robot mechanism, along with its sectional view.

The remaining paper is organized as follows. In Section 2, the compliant modular robot mechanism is introduced, and its quasi-static analysis is provided. Section 3 introduces wheel-slip and presents a systematic procedure to perform wheel-torque optimization to minimize slippage. Section 4 analyzes the optimization results in order to choose the best objective function. In Section 5, a controller is

https://doi.org/10.1017/S0263574715000685 Published online by Cambridge University Press

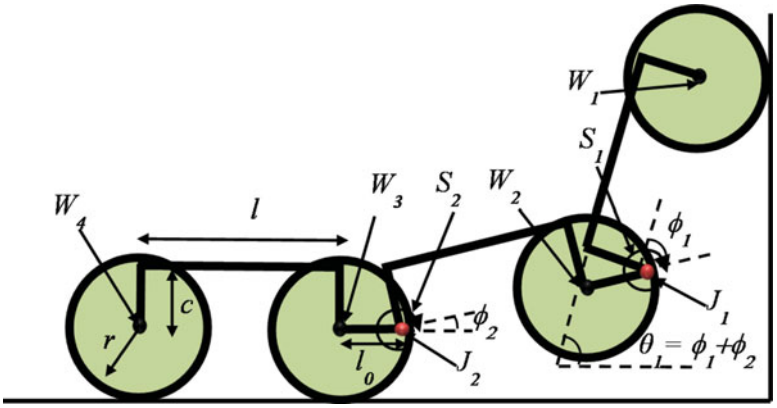


Fig. 2. Schematic of the 3-module robot.

Table I. Specifications of the robot.

Quantity	Symbols	Values(with Units)
Link length	l	0.15 m
Wheel radius	r	0.03 m
Wheel and link joint offset	l_0	0.03 m
Stall torque of wheel motors	τ_{wmax}	0.6 Nm
Stiffness of the spring at J_1	k_1	0.04 Nm/rad
Stiffness of the spring at J_2	k_2	0 Nm/rad
Mass of each wheel	m_w	0.1 Kg
Mass of each link	m_l	0.3 Kg
Mass of the robot	m	1.7 Kg

developed to achieve the desired optimal wheel-torque. Numerical and Experimental results validating the effectiveness of this controller are also provided. Finally, Section 6 contains the conclusions and outline for future work.

2. Model Description

Figure 2 shows a schematic of the proposed robot mechanism consisting of 3 links and 4 wheel-pairs. The link-joints are positioned at the same height as that of the wheel axes with an offset(l_0) equal to the wheel's radius. The joints at wheels and links are denoted by $W_1 - W_4$ and $J_1 - J_2$, respectively. The location of the passive springs is denoted by S_1 and S_2, respectively. Finally, ϕ_i is the relative joint angle between links i and $i + 1$ and θ_i be the absolute joint angle of link i with respect to the ground.

The springs at S_1 and S_2 have stiffnesses 0.04 Nm/rad and 0, respectively. (Joint J_2 is fully passive). A more comprehensive discussion on the robot's compliance design is given in ref. [2]. A 90° double torsion spring is designed for this purpose. It is close wound and having tangent legs. It is fitted into the joint J_1 with zero preload. The robot's specifications are provided in Table I.

Design of an appropriate ground clearance can help in avoiding any undesirable collision between the trunk and obstacles while climbing, as shown in Fig. 3(a). Figure 3(b) shows the minimum clearance required, denoted as c_{min}, for avoiding collision of trunk with step. It is defined as $c_{min} = c + r$, where $c = l/2 - r\sqrt{2}$, l is length of the module and r is the wheel radius. Note that, the clearance also depends on the shape of the obstacle, and it increases with increase in the sharpness of the obstacle/step, i.e., step angle $\leq 90°$ (Fig. 3(b)). In this paper, we focus on climbing obstacle/step with step-angle $\geq 90°$.

2.1. Climbing analysis with a passive modular robot

Figure 4(a) shows the climbing phase of the robot with fully passive joint J_1 (no spring). Note that module 1 will continue to climb along the step till it crosses a limiting angle. Beyond the

https://doi.org/10.1017/S0263574715000685 Published online by Cambridge University Press

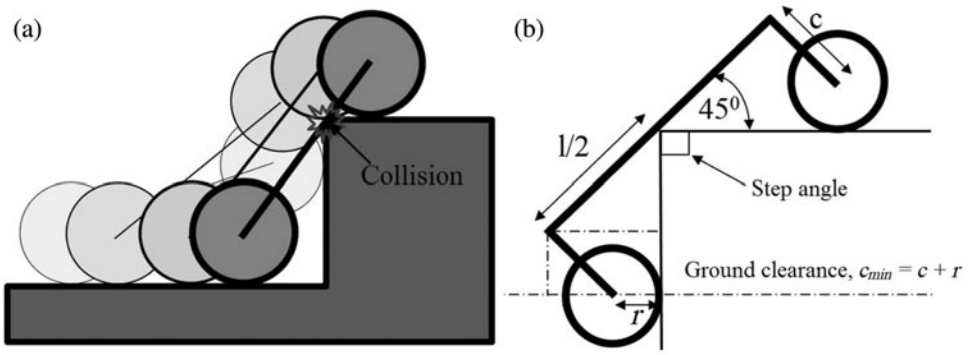


Fig. 3. Effect of ground clearance: (a) An inevitable collision occurs when the module is designed with insufficient ground clearance; (b) The minimum ground clearance c_{min} is parametrized in terms of the length of the module(l) and the wheel radius(r).

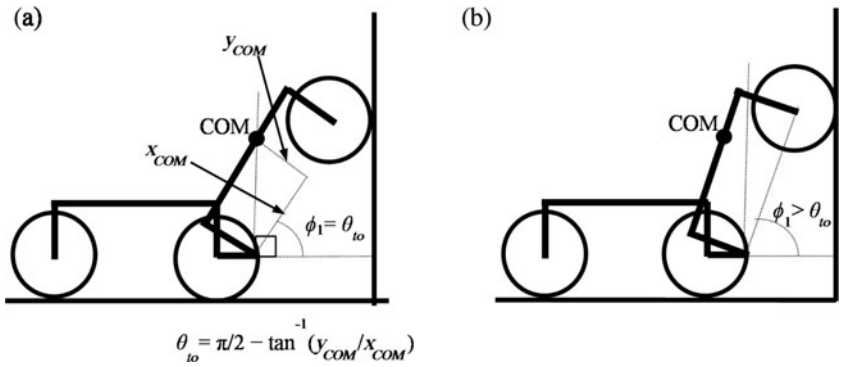


Fig. 4. Climbing behavior of the passive robot: (a) In the climbing phase, it continues to climb the step as long as $\phi_1 \leq \theta_{to}$. (b) If the module continues to climb beyond this point, then the moment due to its self-weight changes direction and causes the module to tip over. This phase is called tip over phase.

limiting angle the module will tip-over as shown in Fig. 4(b). The limiting angle called tip-over angle (θ_{to}) (Fig. 4(a)) can be determined based on the position of center-of-mass (COM) of the module as $\theta_{to} = \pi/2 - tan^{-1}(y_{COM}/x_{COM})$, where x_{COM} and y_{COM} denote the COM coordinates of the module in the module fixed frame. This tip-over phenomenon limits the climbing ability of the proposed robot, and the robot can only climb obstacles of heights less than or equal to $lsin(\theta_{to})$. In our previous work,[2] was shown how adding an optimally designed torsional spring at joint J_1 helped avoided θ_1 crossing θ_{to} by lifting module 2 off the ground. This causes ϕ_1 to decrease while keeping θ_1 nearly constant. This enables the robot to climb higher without having to tip-over.

The aim of this paper is to only use the traction forces developed at the wheel-ground contacts to climb steps. The passive link-joints allow the mechanism to freely deform along an obstacle. For the experiments in this work, the proposed compliant modular robot traverses at a constant speed of 18 cm/s. At such low speeds the dynamic effects are minimal and quasi-static analysis gives a good overview of the forces acting on the robot. Wheel-torques typically are responsible for balancing these external forces and maintain static equilibrium. As the robot is symmetric about the sagittal plane, a planar analysis provides a good approximation of its performance while climbing steep obstacles (assuming that they are also symmetric about the sagittal plane). Static stability equations of the robot are derived in the next subsection.

2.2. *Quasi-static model of the modular robot*

A generic set of static equilibrium equations cannot be employed for analyzing and optimizing the modular robot's step climbing maneuver as it was shown in ref. [10]. In the PAW[10] robot and other passive suspension robots like SHRIMP[11] and CRAB,[8] wheels always maintained contact with the ground. But in the case of the proposed robot, wheel-pairs may lift off the ground, where necessary, to avoid tipping over. This slightly complicates the quasi-static analysis as the static equilibrium

https://doi.org/10.1017/S0263574715000685 Published online by Cambridge University Press

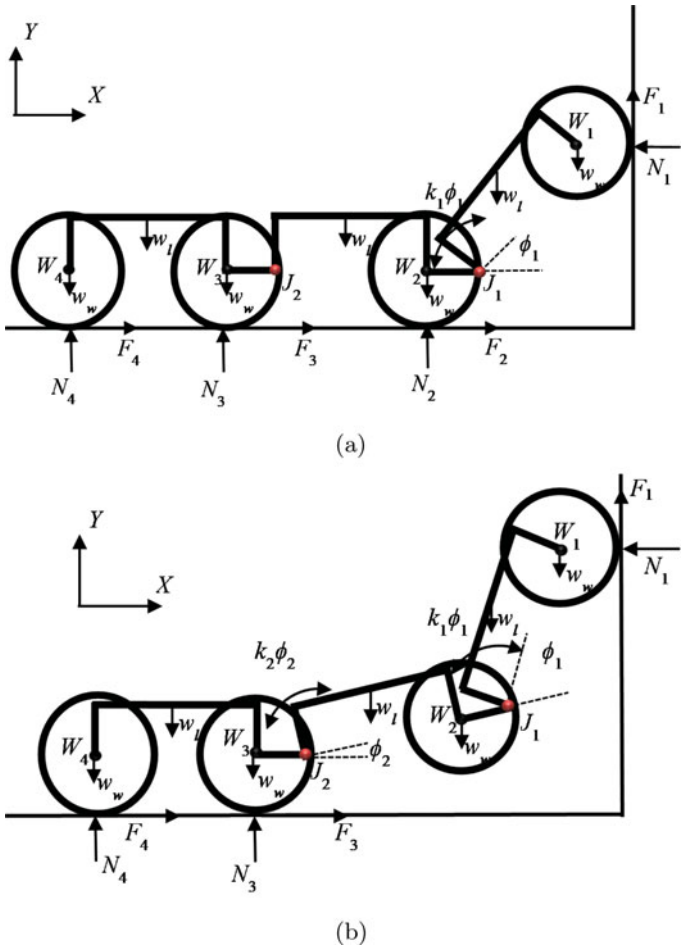


Fig. 5. Snapshots showing the various forces and moments acting on the robot during the 2 climbing phases. (a) Phase-1: one link climbing at any instant. (b) Phase-2: two links climbing at any instant.

equations change when there is a phase change, during the climbing maneuver. Therefore, the robot's static model is divided into two phases, as shown in Fig. 5.

This division is essential as the forces and moments acting on the robot change from one phase to the other. Every time a wheel-pair lifts off the ground, its corresponding normal and traction forces are lost and an additional counter moment is generated due to the spring at the corresponding link-joint. This changes the $\sum F_y$ (net force acting in the y-direction), $\sum F_x$ (net force acting in the x-direction) and $\sum M$'s (net moments about J_1 and J_2), which have to be appropriately adjusted to maintain static equilibrium. Therefore, when every subsequent wheel is lifted off the ground, the robot transits from one climbing phase to the other. Equations (1)–(5) and (6)–(10) contain the minimal set of static-equilibrium equations for the first and second phases of climbing of the robot as shown in Figs. 5(a) and (b), respectively.

$$\sum F_x = 0 \qquad N_1 - F_2 - F_3 - F_4 = 0 \tag{1}$$

$$\sum F_y = 0 \qquad 3w_l + 4w_w - F_1 - N_2 - N_3 - N_4 = 0 \tag{2}$$

$$\sum M_{J_1} = 0 \qquad F_1(l\cos\phi_1 + r) + N_1 l\sin\phi_1 - w_l[(l/2)\cos\phi_1 \text{-} c\sin\phi_1] - k_1\phi_1 - w_w l\cos\phi_1 = 0 \tag{3}$$

$$\sum M_{J_2} = 0 \qquad F_2 r + N_2 l - w_w l - w_l(l/2) - [w_l + w_w - F_1](l + l_0) - k_2\phi_2 + k_1\phi_1 = 0 \tag{4}$$

https://doi.org/10.1017/S0263574715000685 Published online by Cambridge University Press

$$\sum M_{W4} = 0 \quad F_3 r + N_3 l - w_w l - w_l(l/2) - [2(w_l + w_w) - F_1 - N_2](l + l_0) + k_2\phi_2 + F_4 r = 0 \tag{5}$$

$$\sum F_x = 0 \quad N_1 - F_3 - F_4 = 0 \tag{6}$$

$$\sum F_y = 0 \quad 3w_l + 4w_w - F_1 - N_3 - N_4 = 0 \tag{7}$$

$$\sum M_{J_1} = 0 \quad F_1(lcos(\phi_1 + \phi_2) + r) + N_1 lsin(\phi_1 + \phi_2) - w_l[(l/2)cos(\phi_1 + \phi_2) - csin(\phi_1 + \phi_2)] - k_1\phi_1 - w_w lcos(\phi_1 + \phi_2) = 0 \tag{8}$$

$$\sum M_{J_2} = 0 \quad [F_1 - (w_l + w_w)](l + l_0)cos\phi_2 - w_l(l/2cos\phi_2 - csin\phi_2) - w_w lcos\phi_2 + k_1\phi_1 - k_2\phi_2 + N_1(l + l_0)sin\phi_2 = 0 \tag{9}$$

$$\sum M_{W4} = 0 \quad F_3 r + N_3 l - w_w l - w_l(l/2) - [2w_l + 2w_w - F_1](l + l_0) + k_2\phi_2 + F_4 r = 0. \tag{10}$$

These equations are obtained from by simplifications and substitutions of the force-moment equations given in Appendix A. In the first phase, only link 1 is climbing while the other links (and wheels) are on the ground supplying the required push force. Similarly, in the second phase, links 1 and 2 are involved in climbing at any instant. It is expected that the robot can climb heights up to $lsin\theta_{to}$ with 1 climbing link and between $lsin\theta_{to}$ and $2lsin\theta_{to}$ using 2 links. In this paper, the analysis is limited to heights requiring at the most 2 climbing links, thus ensuring that there are always 2 wheel-pairs on the ground to provide sufficient traction at any given point.

It can be seen that there are five equations per phase. The first two equations ensure the equilibrium of forces in the x and y directions, respectively. The remaining equations denote the moment equilibrium at the 2 link-joints J_1 and J_2, and a wheel joint W_4, respectively. Note that, $w_w = 2m_w g$ and $w_l = m_l g$ where m_w and m_l denote the masses of the wheel and link, respectively. F_i and N_i are the traction and normal forces acting on the ith wheel, respectively. k_i is the spring constant of the spring acting at the ith link-joint to maintain static equilibrium at an arbitrary configuration.

The optimal spring stiffness values were obtained for climbing big step-like obstacles while avoiding tip-over in ref. [2]. Only one spring is required at joint J_1 to climb a height of up to 17 cm. Its spring constant is given in Table I. With the addition of springs at the link-joints, this modular robot can successfully climb the rise of a step up to twice its link length with the help of only wheel traction. However, wheel-torques can be used to control this step climbing only under the assumption that there will be no slippage. So this passive compliant robot design cannot be fully exploited without ensuring that the wheel-slip is minimized. In the next section, the causes of wheel-slip are discussed and an optimization formulation with an objective to avoid slippage is described. It was shown in[2] that for a given height h, an n-modular compliant robot with m compliant joints could be developed. As an extension, it can be ascertained that such a robot will have $m + 1$ climbing phases. Without loss of generality, this method can be easily extended to determine optimal wheel-torques of the robot's $n + 1$ wheel-pairs. In the next section, the wheel-slip problem is introduced and potential solutions are suggested.

3. Wheel-Slip Reduction

Figure 6 shows the free-body diagram of a wheel rolling on a flat surface. It can be noted that, the wheel is in static equilibrium when $F = \tau/r$, where r and τ are the wheel's radius and torque, respectively. For the wheel to maintain pure rolling, the frictional force F should always satisfy the friction constraint equations, $F = \mu_s N$, where μ_s is the coefficient of static friction. Here, F is directly proportional to the wheel-torque τ. If the wheel-torque exceeds $\mu_s N$, then the wheel begins to slip and frictional force, $F = \mu_d N$, where μ_d is the coefficient of dynamic friction. The aim is to always keep the wheel-torque within the limit of $\mu_s N$ to avoid any slippage. However, this is very hard to achieve as μ for the wheel-ground interface is not known in advance, and it also changes

https://doi.org/10.1017/S0263574715000685 Published online by Cambridge University Press

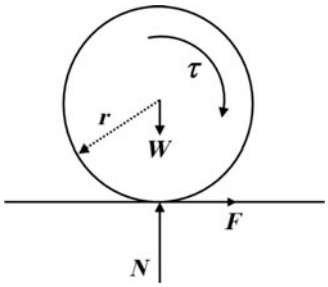


Fig. 6. Free body diagram of the wheel.

dynamically on any terrain. Well-known slip reduction technologies like Anti-Lock Braking System (ABS), which is used extensively in automobiles, are equipped with sensors to detect wheel-slip and a corrective torque control action is taken till the wheel stops slipping. But this method requires slip to occur. A preventive method, which is not only robust but also exploits the robot's suspension mechanism to minimize slippage is desirable. Note that, slippage does not hinder the climbing ability of this robot. The robot can still successfully climb even if there is slip in some of its wheels. However, slip makes it difficult to exercise control on the robot and also expends energy. Therefore, it is very important to study it carefully and create mechanisms which can minimize slip.

To build this analysis, one may first begin by assuming that there is no slip, and then estimate the F_i and N_i values across various points on the terrain by minimizing $\sum_i F_i/N_i$ to achieve the objective of no slip. Denote the maximum value of the F_i/N_i ratio obtained from this optimization as μ_o. The mechanism with lower μ_o can traverse on a wider range of surfaces (whose μ_s w.r.t the wheel is between μ_o and 1) without slipping. This is the most intuitive objective function for the optimization procedure, as shown in Eq. (11). However, since the function is non-linear, it may get stuck in local optima. Also note that, this objective function does not enforce all the wheel-pairs to maintain the same μ_o value. This can be enforced by using Eq. (12) instead. Though these two objective functions best capture the objective of this optimization procedure, they both are non-linear and may get stuck in local optima. The objective can be linearized by just minimizing the sum of traction forces, as shown in Eq. (13), instead of minimizing the sum of traction-to-normal ratios. Since this function is linear it is guaranteed to converge to a global optimum value. The relative merits of these three objective functions is further analyzed in the subsequent sections and the best one will be chosen to develop the optimal wheel-torque based controller.

$$\sum_{i=1}^{4} F_i/N_i \tag{11}$$

$$\sum_{i=1}^{4}(F_i/N_i - \mu_{avg})^2 \quad where,\ \mu_{avg} = \sum_{i=1}^{4} F_i/N_i/4 \tag{12}$$

$$\sum_{i=1}^{4} F_i \quad . \tag{13}$$

Before executing the optimization procedure, the constraint equations need to be derived. Constraint equations capture the static stability requirement of the robot at any intermediate configuration while step-climbing. To this end, in the next subsection, a method is introduced to determine the robot's configuration at any given height.

3.1. Posture estimation

The use of 2-links for climbing at any instant, allows the robot to climb steps/walls as high as $2lsin(\theta_{to})$, i.e., h_{max}, without tipping over as explained in Section 2.1. To estimate the postures, the step height is divided into n set points between 0 and h_{max}, each point separated by the distance of 0.01 m. The posture of the robot at all the set points is determined so that they can be used

https://doi.org/10.1017/S0263574715000685 Published online by Cambridge University Press

to derive the static equilibrium equations for that posture. For step heights that involve only one climbing link, i.e., heights between 0 and $lsin(\theta_{to})$ (phase-1), the posture of the robot, in terms of the joint angles, can be determined as $\phi_1^j = sin^{-1}(h^j/l)$ and $\phi_2 = 0$. When $\phi_1 = 70^\circ$ ($phi_1 \approx \theta_{to}$), i.e., after having climbed a height of 0.14 m, the robot transits to phase-2. Wheel-pair-2 is lifted off the ground and the robot now has two climbing members. This process reduces ϕ_1 and increases ϕ_2 while ensuring that $\theta_1 \leq \theta_{to}$. The desired postures for the second phase can be obtained by running a simple optimization procedure for all the heights between $lsin\theta_{to}$ and $2lsin\theta_{to}$ as shown in Eq. (14). The objective function ensures the net change in the relative angles is minimized while moving from one set-point to the other. This is very important from the quasi-static point of view as a drastic change in posture with a small increase in height cannot be attained without taking the dynamics of the system into consideration.

$$\begin{aligned} \underset{\phi_1,\phi_2}{\text{minimize}} \quad & \sum_{i=1}^{2} \left(\phi_i^j - \phi_i^{j-1}\right)^2 \\ \text{subject to} \quad & lsin(\phi_1 + \phi_2) + lsin(\phi_2) = h^j \\ & \phi_1 + \phi_2 \leq \theta_{to} \\ & 0 \leq \phi_1, \phi_2 \leq \theta_{to}. \end{aligned} \tag{14}$$

This provides the posture desired from the robot at various set-points. The objective functions have to be minimized across all these set-points to study the robot's performance and localize potential regions for slippage.

3.2. Optimization routine

The design variables are F_i's and N_i's of all the four wheels. For maintaining an arbitrary posture of the robot, the static equilibrium equations have to be satisfied at that posture. Thus, they form the equality constraints to this problem. These equations change from one phase to the other at the set point $lsin\theta_{to}$, to reflect the fact that the second wheel is lifted off the ground. Therefore, for phase-1, the equality constraints are obtained from Eqs. (1)–(5) and for phase-2, it is obtained from Eqs. (6)–(10). As the posture is predetermined for a given height, the equality constraints are linear. However, since some of the objective functions are non-linear, standard functions like $fmincon$ (of MATLAB) rely on a strong initial guess provided by the user to search for the globally optimal value. Providing a good initial guess is hard in this scenario as the optimization routine has to be run on all the set points and each one may have a good initial guess of its own. An alternative approach is to further compress the feasible region by tightly bounding the design variables using the knowledge of their properties and the desired objective. To this end, additional linear inequality constraints are added to the system to reduce the feasible region and ensure that the proximity of the obtained solution is as close to the global minimum as possible. Equation (15) is to restrict the optimal F/N ratio to always remain between 0 and 1 for all the wheels and at all the set points. Equation (16) bounds the wheel motor torques for all the wheels with their maximum values. Finally, the ratio will also decrease if normal N increases. Therefore, equations in Eq. (17) ensures that the search space consists of only regions where N increases or maintains atleast N_{avg}, viz. the normal force on a flat terrain. Additionally, for phase 2, F_2 and N_2 are equated to 0 as wheel 2 is lifted off the ground and its traction no longer contributes to step climbing.

$$F_i \leq N_i \quad \forall\, i = 1 \ldots 4 \tag{15}$$

$$0 \leq F_i \leq \tau_{wmax}/r \quad \forall\, i = 1 \ldots 4 \tag{16}$$

$$\begin{aligned} phase\ 1: \quad & N_i \geq N_{avg} \qquad i \in \{2, 3, 4\} \\ phase\ 2: \quad & N_i \geq N_{avg} \qquad i \in \{3, 4\}. \end{aligned} \tag{17}$$

https://doi.org/10.1017/S0263574715000685 Published online by Cambridge University Press

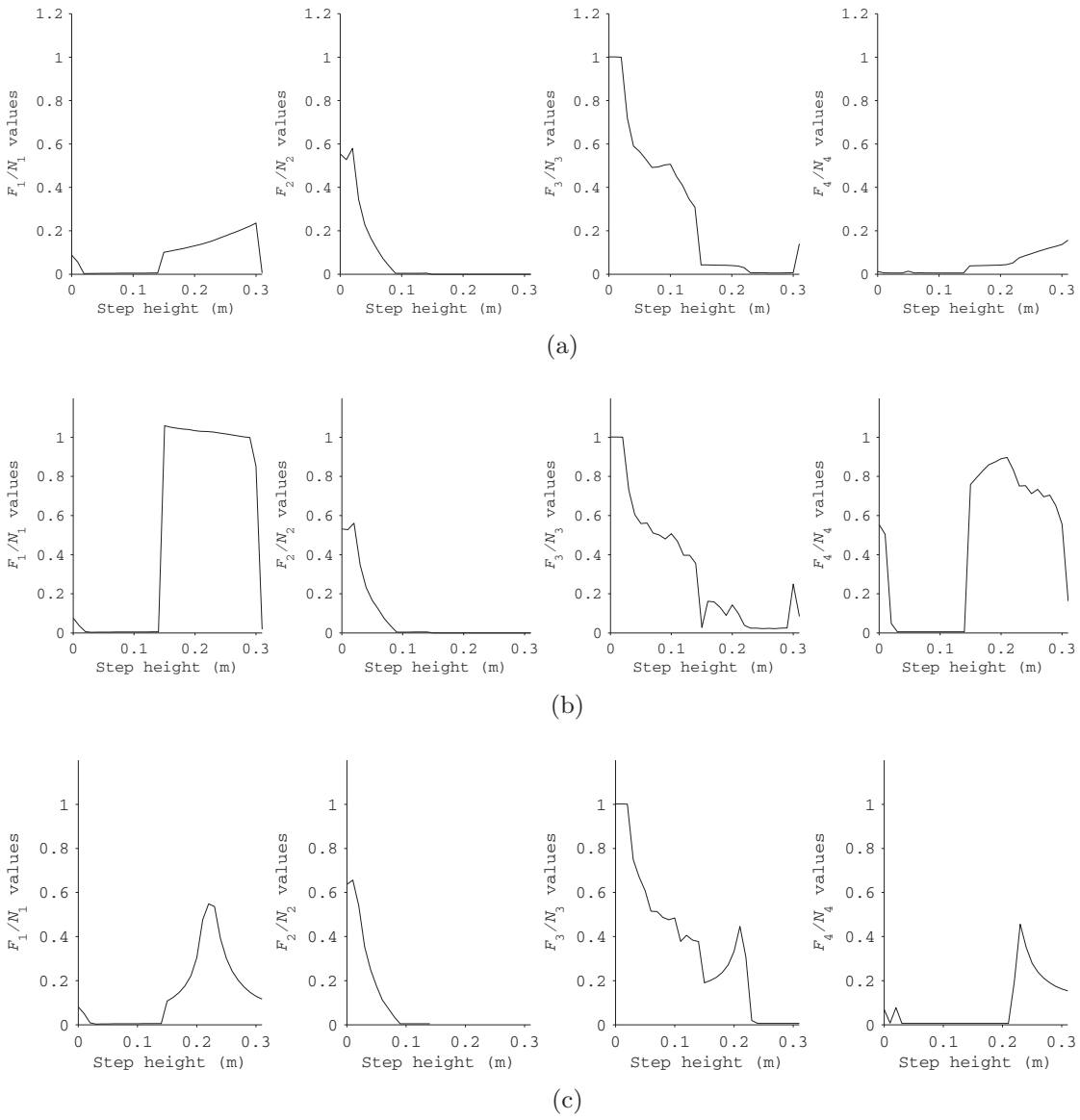


Fig. 7. Optimized F/N values for all the four wheel-pairs using the three objective functions. (a) Objective-1. (b) Objective-2. (c) Objective-3.

The above mentioned optimization routine is performed for step of height 0.310 m and at all its intermediate set points. The results are presented and discussed in the next section.

4. Optimization Results and Discussion

Wheel-torque optimization is carried out using the three objective functions as given in Eqs (11)–(13) at all the set points between 0 and h_{max}. The traction-to-normal force ratios for all the wheel pairs are plotted against the height set points, as shown in Fig. 7. It can be seen that none of the objective functions yield F/N values that are consistent across heights and wheel-pairs. Interestingly, Objective-2 maintained consistency while in a climbing phase, but it changes sharply when it transits from one climbing phase to another.

During the initial stages of climbing, $F_3/N_3 \approx 1$ in all the three cases. This implies that wheel-pair-3 will most likely slip as the robot begins to climb. Note that, μ_o alone an inefficient metric to assess the performance of the robot. Therefore, two additional metrics are used, namely Mean and Mode of the traction-to-normal force ratios for all the wheel-pairs, across height set-points, per objective

https://doi.org/10.1017/S0263574715000685 Published online by Cambridge University Press

Table II. Performance metrics for each objective function.

Objective Function	Max. value (μ_o)	Mean	Mode
Objective-1	1.0017	0.1464	0.0023
Objective-2	1.0593	0.3887	0.0030
Objective-3	1.0018	0.1926	0.0029

function. The values thus obtained are listed in the Table II. It is very counter-intuitive to note that objective-2 (given by Eq. (12)) performs poorly on all the metrics. This could be due to the highly non-linear nature of the function, making it most susceptible to settling at local minima. It is equally interesting to note that objective function-3 (given in Eq. (13)) is also not the best objective function in spite of being linear and having global optima guarantees. Objective-1 (given by Eq. (11)) performs consistently well on all the metrics. It may be noted that Objectives-1 and 3 perform very similarly in climbing phase-1. However in phase-2, objective-1 performs much better, as seen in Figs. 7(a) and (c). This can also be verified from the optimal wheel-torque plots shown in Fig. 8. It can be clearly seen that wheel-torques in Fig. 8(c) may saturate sooner than those in Fig. 8(a). Thus, Objective-1 is finalized as the desired objective function. We now focus the optimal traction-to-normal force ratios and optimal wheel-torques obtained using objective-1 to further analyze its behavior.

It can be noted from Fig. 7(a) that, for wheel-pair-1, the F/N ratio is consistently lower (max $F_1/N_1 = 0.2357$) as compared to all other wheels and especially lower in phase-1. In phase-1, all the three wheel-torques combine to provide the horizontal normal force N_1. Thus, N_1 is always greater than or equal to $3N_{avg}$. Even though the required wheel-torque for rolling might remain the same while climbing the step, the normal force has more than tripled thus greatly reducing F_1/N_1. This implies that this robot can climb on a very slippery surface without slipping as the reduction in μ_s is being compensated by increase in normal force N. This is the key novelty of modularity that this robot design wishes to exploit. For phase-2 however, the normal force $N_1 \geq 2N_{avg}$. Therefore, this robot can climb a step without slipping even if μ_s drops to a third of its original value in phase-1 or to half its value in phase-2. This makes robot mechanism robust to changes in μ_s during step climbing.

For wheel-pair-2, F_2/N_2 reduces with an increase in height in phase 1. However, in phase-2, wheel-pair-2 is not actuated as it can no longer provide traction. For wheel-pair-3, as shown in Fig. 7(c), a trend similar to that of wheel-pair-2 is observed. It starts decreasing appreciably in phase-2 when wheel-pair-2 is off the ground as N_3 increases. Wheel-pair-4 has a counter-intuitive trend. The average F_4/N_4 value is 0.0483 which is very low. However, the ratio increases in phase-2 instead of decreasing as in the case of other wheels. When the second wheel-pair is lifted off the ground, to maintain static equilibrium in the x-direction, forces are redistributed thus increasing the values of F_3 and F_4. However, for wheel-pair-3, N_3 also increases accordingly and therefore the ratio could be kept lower. The normal force N_4, on the other hand, does not increase proportionally thus increasing the ratio in the case of wheel-pair-4.

Note that, during phase-1, $F_1/N_1 = 0$ and $F_4/N_4 = 0$. This implies that, in the optimal case, wheel-pairs-1 and -4 do not apply any traction to maintain static-stability. In effect, wheel-pairs-1 and -4 are simply rolled along without needing any actuation. As the robot begins to climb up, wheel-pair-1 loses its normal force in the vertical direction and subsequently N_2 goes up to do weight balance. Therefore, intuitively, the objective function is keeping the F/N values low by driving F_1 to zero and making wheel-pair-2 (thus F_2) take the additional load. Since, wheel-pair-4 normal force gets least effected during this process, F_4/N_4 is minimized by driving F_4 to zero. Therefore, in phase-1, wheel-pairs-2 and -3 play a major role in maintaining static balance. However, in phase-2, wheel-pair-2 is lifted off the ground and this forces wheel-pairs-1 and -4 to start exerting least possible force to maintain static balance while keeping their respective F/N ratios low.

Figure 8(a) shows the corresponding desired wheel-torque values for all wheel-pairs obtained using Objective-1. Note that, the torque values are quite low in phase-2 as compared to phase-1. This is attributed to spring action. In phase-1, a part of the wheel-traction generated was used to balance the resisting moment generated by the spring. However, in phase-2, the spring deformation is much lower. Additionally, in spite of their low values, the moments due to N_1 and F_1 are high enough to balance moments due to weight at J_2 as the moment arm length increased significantly.

https://doi.org/10.1017/S0263574715000685 Published online by Cambridge University Press

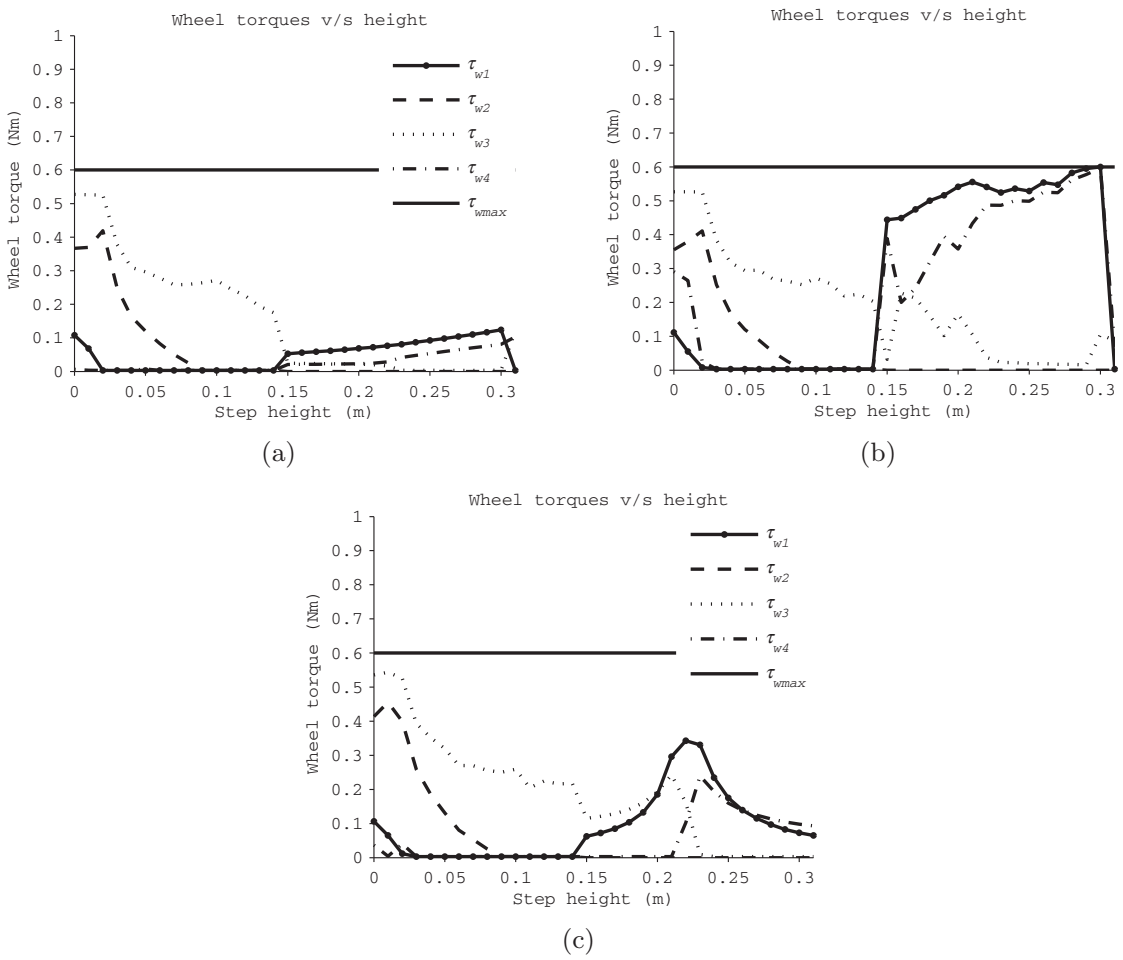


Fig. 8. Optimal wheel-torque plots for all the wheel pairs with (a) objective function-1, (b) objective function-2 and (c) objective function-3.

The optimal wheel-torques obtained so far only maintain the robot in static equilibrium at a given height. However, the robot has to be imparted with some motion to move from one set-point to another. Additional torque needs to be applied to move the robot with a desired velocity. Therefore, a velocity controller is coupled to these optimal wheel to obtain the desired optimal wheel-torque controller as described in the next section.

5. Proposed Wheel-Torque Controller Design

Optimal wheel-torque control is a very active research area and several instances of successful application of wheel-torque control for mobile robots, like SHRIMP[11] and CRAB,[6] have been reported earlier in refs. [4], [5], [7], [8]. A detailed study of this method is given in ref. [12].

The discrete optimal-wheel-torque values obtained from the previous section are fitted using cubic splines such that they can be approximately estimated for any arbitrary value of θ_1. For the height that the robot climbed during the experiments, the robot's state can be quite accurately determined using θ_1. The implementation details will be discussed in the next sub-section. In this manner, the optimal-torque values are determined at every time-step and supplied to the robot in an open-loop. While this maintains static stability at any given time-step, the robot needs to be imparted some velocity to traverse from one state to another. To this end, a desired velocity, V_d, is chosen. Accordingly, a velocity Proportional-Derivative Controller is developed. The torques required to maintain static-stability (optimal-wheel-torques) and to maintain speed are then added and supplied to the robot. This combination will be referred to as torque controller, hereafter. This velocity control loop ensures that the robot never gets stuck in a statically stable configuration, during the step ascent. The block

https://doi.org/10.1017/S0263574715000685 Published online by Cambridge University Press

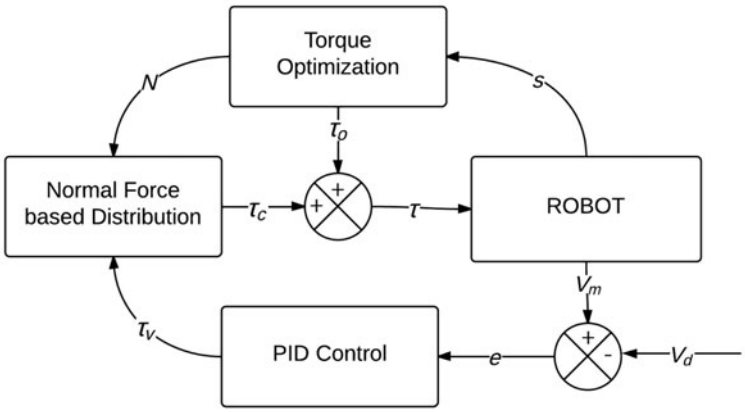


Fig. 9. Block diagram of the proposed controller.

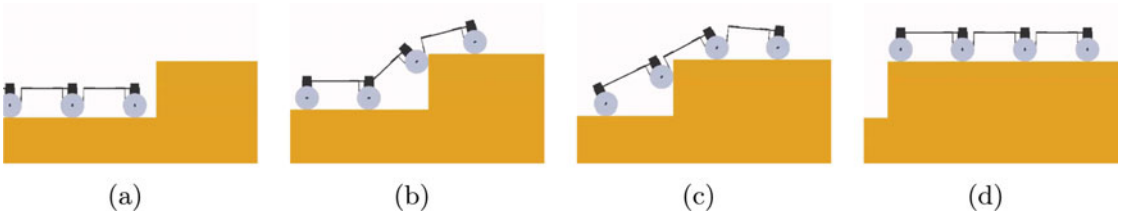


Fig. 10. Snapshots of the simulation.

diagram of the proposed controller is shown in Fig. 9, where s denotes the state of the robot, V_m and V_d denote the measured and desired wheel velocities, e denotes the velocity error term, and τ_o, τ_v and τ_c denote the torque values obtained from the optimizer, velocity controller and torque scaled according to normal force, respectively.

It is worth noting that the normal forces (N) are used to augment the application of τ_v to all the wheels, as shown in Fig. 9. Sometimes, PD controller commands high torques to quickly reach a desired speed value. However, if the normal force at the wheel at that moment is very low, it will cause the wheel to slip, while reaching the desired velocity. To minimize this problem, the desired τ_v's are scaled down based on the normal force values at that state, as obtained from the optimization procedure. This way, the wheels can apply a higher fraction of τ_v when the normal forces are higher too. This also ensures that $\tau_{v_i} = 0$ when the ith wheel loses contact with the ground, thus saving energy. In Fig. 9, τ_c denotes the corrected torque for each individual wheel for achieving the desired velocity. Note that, while this process may slow down a wheel from achieving the desired velocity, it still achieves the larger goal of generating forward motion with minimal slip.

5.1. Numerical and experimental results

Two key metrics were used to study the performance of the robot. They are average slip ratio(s_a) and mean wheel-torque ($\overline{T}$). Slip ratio is given by, $1 - v/r\omega$, where v is the linear velocity of each wheel and ω is the angular velocity measured by wheel encoder. The slip ratios of all the wheels were determined and averaged to obtain the average slip ratio. Similarly, the wheel-torques for all the wheels were measured and averaged to obtain the mean wheel-torque $\overline{T}$.

The utility of the proposed torque control was compared to the conventional velocity control, and the net reduction in slip ratio and torque requirement were estimated. Numerical experiments were conducted in a multibody dynamics software (MSC ADAMS), as shown in Fig. 10.

Simulations were carried out in which the robot was made to climb a step of height 14 cm. Note that the maximum wheel-slip occurs in the robot while climbing the rise of a step. Therefore, the average slip ratio and mean torque requirement were measured for all the wheels during the step climbing phase. The coefficient of friction μ of the wheel-ground contact was varied between $0.6 - 1.0$ and several test runs were taken to study the climbing behavior of the robot. ADAMS' built-in measure functions were used to determine angular velocity, linear velocity and torque readings for all the

https://doi.org/10.1017/S0263574715000685 Published online by Cambridge University Press

Table III. Results for average slip ratio and mean torque.

	Average Slip Ratio			Mean wheel-torque (Nm)		
μ value	Velocity Control	Torque Control	% reduction	Velocity Control	Torque Control	% reduction
0.6	0.046	0.040	13.71	0.084	0.077	8.95
0.8	0.035	0.027	22.37	0.086	0.073	15.03
1	0.029	0.021	26.04	0.088	0.075	15.23

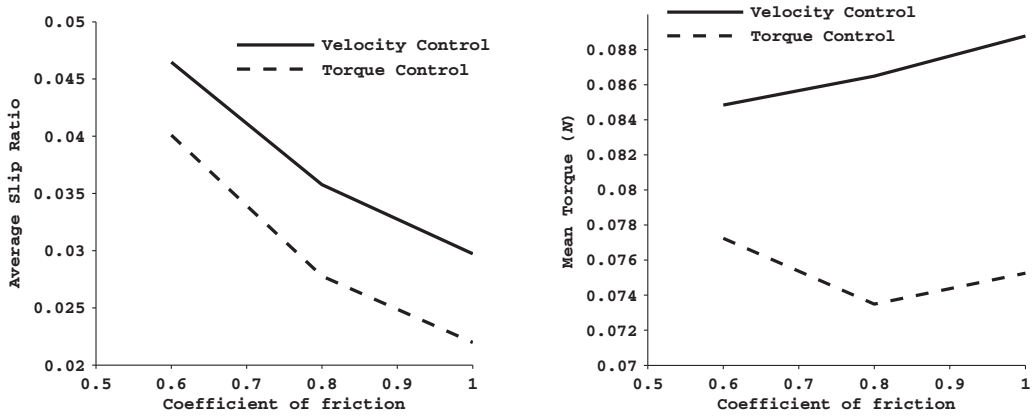


Fig. 11. Plots for average slip ratio and mean torque, with μ varying between $0.6 - 1.0$, using both velocity control and torque control.



Fig. 12. Snapshots of the robot climbing, a 30 cm high step.

wheels. They were used to calculate mean slip ratio and mean torque requirement. The results of mean slip ratio and mean wheel-torque for different values of coefficient of friction are shown in Table III and Fig. 11. It can be seen that there is a considerable reduction in overall wheel-slip when torque control is implemented. There is also a decrease in the average torque requirement.

It is worth noting that the robot climbs the step faster when it uses torque control. There are two reasons for this, firstly, the optimal torques obtained from the optimization procedure ensure that the system is statically balanced. This implies that the velocity control needs to apply only the necessary torque to achieve the desired velocity. Secondly, τ_c enables the wheels with higher normal forces to apply greater traction. It was shown inref. [2] that, due to the spring compliance, the normal force N_1 increases as wheel-pair 1 climbs the rise of the step. Consequently, wheel-pairs will require less time to climb the step, especially wheel-pair 1. This is shown experimentally in Table V. Wheel-pair-1 consistently took less time to climb the step across surfaces.

Torque control was also implemented on the experimental prototype and to further validate the simulation results. The performance of the experimental prototype was tested on various types of wheel-ground interfaces. It was made to climb a step of height 14 cm. It was previously shown inref. [2] that the robot can climb steps up to 17 cm height. Infact, it can climb a step riser which is as high as 30 cm without tipping over, as shown in Fig. 12. However, for the experiments conducted in this paper, a relatively lower height if 14 cm is chosen. Note that, the focus of this paper is to test and verify the utility of wheel-torque optimization in slip reduction for a robot whose climbing ability has already been discussed and documented. Secondly, 14 cm is specifically chosen as it roughly equals

https://doi.org/10.1017/S0263574715000685 Published online by Cambridge University Press

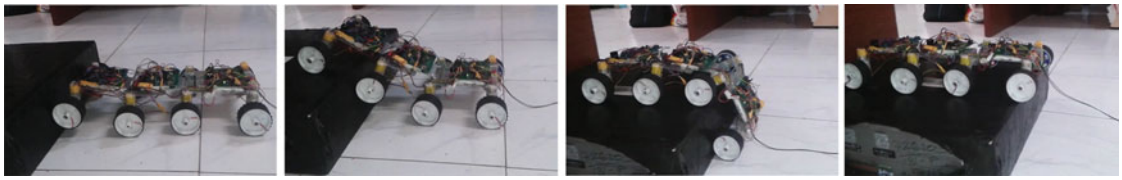


Fig. 13. Snapshots of the robot climbing the tiled floor.

Table IV. Experimental results for average slip ratio and mean torque on three surfaces using velocity and torque controls.

		Mode of control		
Floor type		Velocity control	Torque control	% Reduction
Tiled	Mean slip	0.189	0.052	72.27
	Mean torque	0.087	0.062	28.00
Carpeted	Mean slip	0.204	0.082	59.80
	Mean torque	0.080	0.066	16.82
Taped	Mean slip	0.280	0.237	15.36
	Mean torque	0.080	0.067	15.59
Wet	Mean slip	0.238	0.169	28.99
	Mean torque	0.072	0.070	3.73

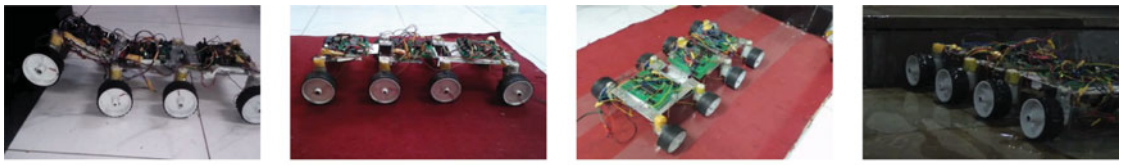
Table V. Average time taken by Wheel-1 to climb the step.

Floor Type	Velocity Control	Torque Control
Tiled	6.20	5.47
Carpeted	5.65	4.41
Taped	5.42	4.84
Wet	6.09	5.36

the height of a step-riser in residential stairways. The proposed compliant robot is being altered to climb stairs and having this torque controller would not only make climbing more energy efficient but also make wheel odometry more accurate.

Each link of the robot is fitted with an IMU sensor to accurately estimate it angle θ_i. The IMU readings obtained from the first link are sent to the torque controller to estimate optimal torque. The controller is implemented on a custom designed AtMega64 board. The robot was run on four types of surfaces namely, tiled, carpeted, taped and wet, as shown in Figs. 14(a) and (d). Snapshots of the robot climbing the obstacle from a tiled floor are shown in Fig. 13. The average slip ratio and mean torque values obtained for all surfaces are tabulated in Table IV. Angular velocities of individual wheels were noted using a wheel encoders. Starting and stopping points were marked for all the wheels. The fixed distance travelled by the wheels was timed and linear velocities were estimated from this data. Slip-ratio for each wheel can be calculated as $\sigma_i = 1 - v_i/r\omega_i$, where v and ω_i denote the linear and angular velocities of the ith wheel, respectively. Therefore, the average slip ratio is equal to, $\sum_{i=1}^{8} \sigma_i/8$. For DC motors, the torque generated is directly proportional to the current supplied. Therefore, individual wheel-torques are determined indirectly by measuring the current supplied to individual motors using a current sensor. The motor-constant was obtained from the motor's datasheet. Mean wheel-torque is obtained by averaging the wheel-torques of all the wheels for the duration of climbing. Five trails were carried out on each surface. Note that, there was a significant reduction in the wheel-slip and mean torque requirement, which validates the simulation results. This shows the efficacy of the proposed approach. As discussed earlier, the time taken to climb also is shorter when torque control is used. This was observed for all the wheels and across all the surfaces. The values for Wheel-pair-1 are provide in Table V.

https://doi.org/10.1017/S0263574715000685 Published online by Cambridge University Press



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

Fig. 14. Experiments on the prototype: The robot was made to climb on, (a) Tiled , (b) Carpeted ,(c) Taped and (d) Wet floors.

6. Conclusions and Future Work

This paper proposes a wheel-torque optimization of a modular wheeled robot with compliant link-joints. The quasi-static analysis of the model is presented, and used to determine optimal spring stiffness values for each link-joint. It is shown in sufficient detail how the robot can negotiate steps whose heights are up to twice its module's link length using only wheel traction. Next, wheel-torque optimization is presented to minimize wheel-slip leading to better control and energy efficient climbing. This also helps in predicting wheel-slip of the robot on wheel-ground contacts whose coefficients of friction can be estimated. Finally, a wheel-torque controller is developed based on this optimization that leads to considerable slip reduction when compared to conventional Velocity-PID controllers.

A similar wheel-torque optimization for climbing down motion will be investigated in the future. Some of the more intuitive objective functions used in this work are fractional. Solving them using conventional non-linear optimization solvers often yield only sub-optimal solutions. There are special formulations and solvers that exploit the structure of fractional problems (sum of linear ratios, fractional programming, etc.) and solve for global optima. They will also be explored in the future.

Appendix A. Static Equilibrium Equations

This appendix presents free-body diagrams for the two climbing phases and related static equilibrium equations. These equations form basis for Eqs. (1)–(5) and (6)–(10).

1. Free Body diagrams for the One Link Climbing Case
   (a) Module 1
   For Wheel-1:

$$\sum F_x = 0: \; fx_{w1} - N_1 = 0 \tag{18}$$

$$\sum F_y = 0: \; fy_{w1} - w_w + F_1 = 0 \tag{19}$$

$$\sum M_{W_1} = 0: \; \tau_{w1} - F_1 r = 0. \tag{20}$$

For Trunk-1:

$$\sum F_x = 0: \; fx_{l1} - fx_{w1} = 0 \tag{21}$$

$$\sum F_y = 0: \; fy_{l1} - w_l - fy_{w1} = 0 \tag{22}$$

$$\sum M_{J_1} = 0: \; k_1\phi_1 + w_l((l/2)cos\phi_1 - csin\phi_1) + fy_{w1}lcos\phi_1 \; fx_{w1}lsin\phi_1 - \tau_{w1} = 0. \tag{23}$$

   (b) Module 2
   For Wheel-2:

$$\sum F_x = 0: \; fx_{w2} - F_2 = 0 \tag{24}$$

$$\sum F_y = 0: \; fy_{w2} - N_2 + w_w = 0 \tag{25}$$

$$\sum M_{W_2} = 0: \; \tau_{w2} - F_2 r = 0. \tag{26}$$

https://doi.org/10.1017/S0263574715000685 Published online by Cambridge University Press

For Trunk-2:

$$\sum F_x = 0: \; fx_{l2} - fx_{l1} + fx_{w2} = 0 \tag{27}$$

$$\sum F_y = 0: \; fy_{l2} + fy_{w2} - fy_{l1} - w_l = 0 \tag{28}$$

$$\sum M_{J_2} = 0: \; k_2\phi_2 - k_1\phi_1 + w_l(l/2) + fy_{l1}(l + l_0) - fy_{w2}l - \tau_{w2} = 0. \tag{29}$$

(c) Module 3
For Wheel-3:

$$\sum F_x = 0: \; fx_{w3} - F_3 = 0 \tag{30}$$

$$\sum F_y = 0: \; fy_{w3} - N_3 + w_w = 0 \tag{31}$$

$$\sum M_{W_3} = 0: \; \tau_{w3} - F_3 r = 0. \tag{32}$$

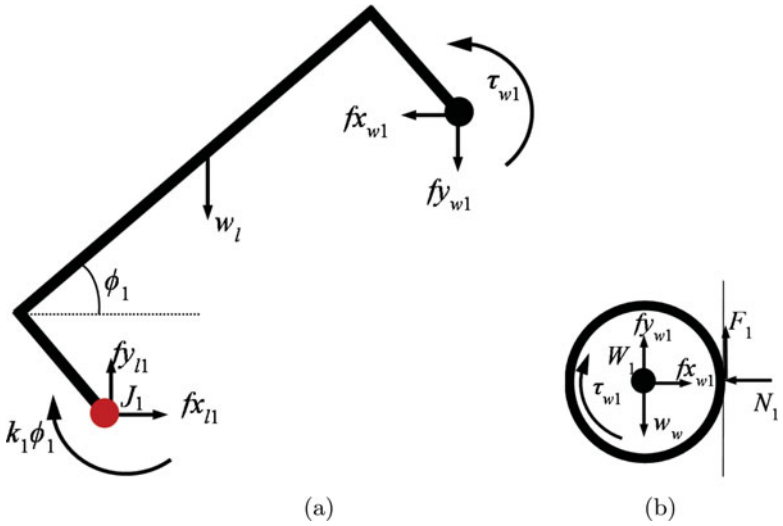


Fig. 15. (a) Trunk 1 (b) Wheel 1.

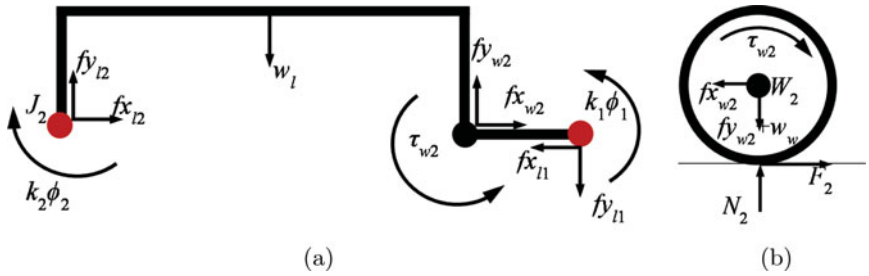


Fig. 16. (a) Trunk 2 (b) Wheel 2.

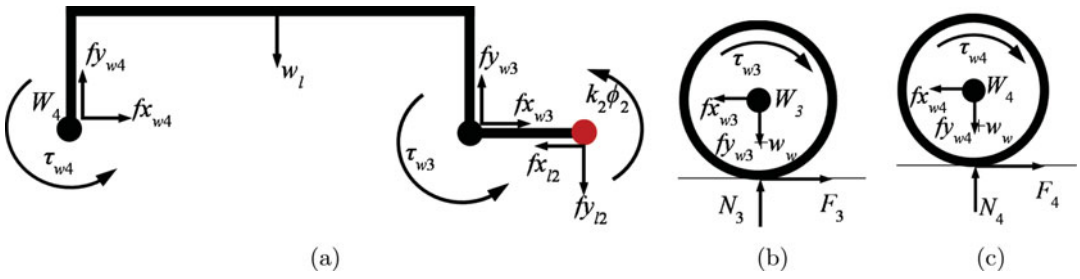


Fig. 17. (a) Trunk 3 (b) Wheel 3 (c) Wheel 4.

https://doi.org/10.1017/S0263574715000685 Published online by Cambridge University Press

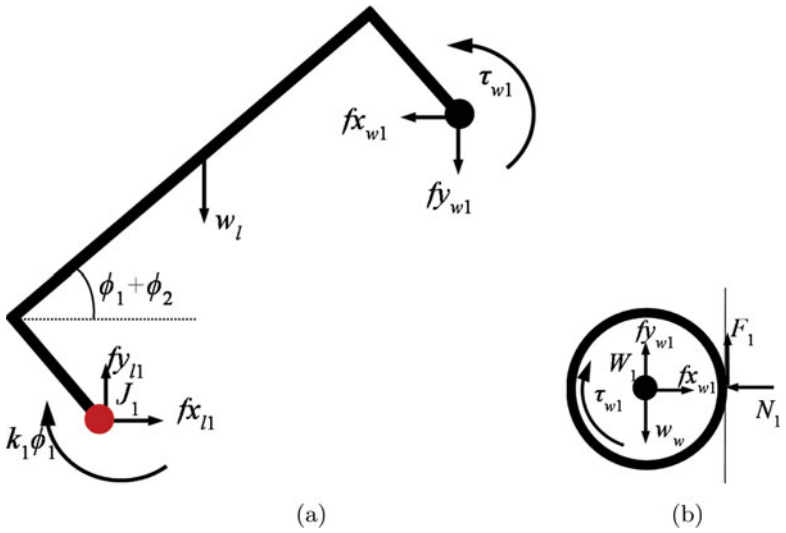


Fig. 18. (a) Trunk 1 (b) Wheel 1.

For Wheel-4:

$$\sum F_x = 0: \ fx_{w4} - F_4 = 0 \tag{33}$$

$$\sum F_y = 0: \ fy_{w4} - N_4 + w_w = 0 \tag{34}$$

$$\sum M_{W_4} = 0: \ \tau_{w4} - F_4 r = 0. \tag{35}$$

For Trunk-3:

$$\sum F_x = 0: \ fx_{l2} - fx_{w3} - fx_{w4} = 0 \tag{36}$$

$$\sum F_y = 0: \ fy_{w4} + fy_{w3} - fy_{l2} - w_l = 0 \tag{37}$$

$$\sum M_{W_4} = 0: \ w_l(l/2) + fy_{l2}(l + l_0) - k_2\phi_2 - fy_{w3}l - \tau_{w3} - \tau_{w4} = 0. \tag{38}$$

(d) Minimal set of equations for phase-1
The minimal set of equations for Phase-1, i.e., Eqs. (1)–(5) are obtained by eliminating the constraint forces (fx_{li}, fy_{li},fx_{wi} and fy_{wi}) from the above equations. The steps are illustrated below:

(i) It can be seen from Eqs. (18), (21), (24), (27), (30) and (33) that $fx_{w1} = N_1$, $fx_{l1} = fx_{w1} = N_1$, $fx_{w2} = F_2$, $fx_{l2} = fx_{l1} - fx_{w2} = N_1 - F_2$, $fx_{w3} = F_3$ and $fx_{w4} = F_4$, respectively. Therefore, Eq. (1) is obtained by substituting the values fx_{l2}, fx_{w3} and fx_{w4} from the above into Eq. (36).

(ii) It can be seen from Eqs. (19), (22), (25), (28), (31) and (34) that $fy_{w1} = w_w - F_1$, $fy_{l1} = w_l + fy_{w1} = w_l + w_w - F_1$, $fy_{w2} = N_2 - w_w$, $fy_{l2} = -fy_{w2} + fy_{l1} + w_l = -N_2 + 2w_w + 2w_l - F_1$, $fy_{w3} = N_3 - w_w$ and $fy_{w4} = N_4 - w_w$, respectively. Hence, Eq. (2) is obtained by substituting the values fy_{w4}, fy_{w3} and fy_{l2} from the above into Eq. (37).

(iii) Equation (3) is obtained by substituting values of fx_{w1}, fy_{w1} and τ_{w1} from Eq. (18)–(20), respectively, into Eq. (23).

(iv) Equation (4) is obtained by substituting values of fy_{l1}, fy_{w2} and τ_{w2} from $item - ii$ above, Eqs. (25) and (26), respectively, into Eq. (29).

(v) Equation (5) is obtained by substituting values of fy_{l2}, fy_{w3}, τ_{w3} and τ_{w4} from $item - ii$ above, Eqs. (31), (32) and (35), respectively, into Eq. (38).

https://doi.org/10.1017/S0263574715000685 Published online by Cambridge University Press

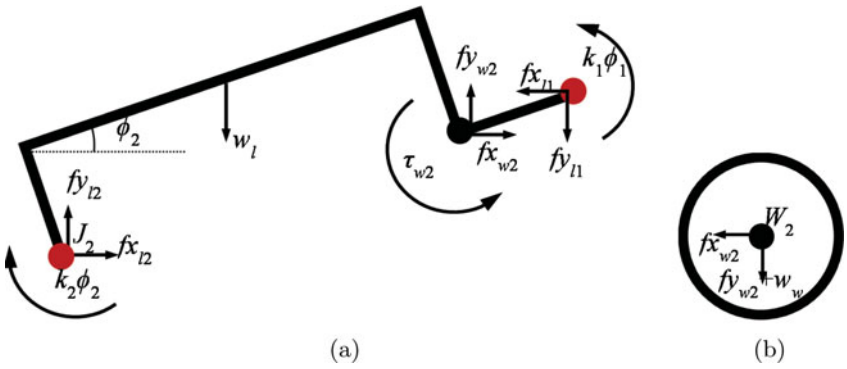


Fig. 19. (a) Trunk 2 (b) Wheel 2.

2. Free Body diagrams for the Two Link Climbing Case
   (a) Module 1
   For Wheel-1:

$$\sum F_x = 0: \ fx_{w1} - N_1 = 0 \tag{39}$$
$$\sum F_y = 0: \ fy_{w1} - w_w + F_1 = 0 \tag{40}$$
$$\sum M_{J1} = 0: \ \tau_{w1} - F_1 r = 0. \tag{41}$$

For Trunk-1:

$$\sum F_x = 0: \ fx_{l1} - fx_{w1} = 0 \tag{42}$$
$$\sum F_y = 0: \ fy_{l1} - w_l - fy_{w1} = 0 \tag{43}$$
$$\sum M_{J1} = 0: \ k_1\phi_1 + w_l[(l/2)cos(\phi_1 + \phi_2) - csin(\phi_1 + \phi_2)] + fy_{w1}lcos(\phi_1 + \phi_2) - fx_{w1}lsin(\phi_1 + \phi_2) - \tau_{w1} = 0. \tag{44}$$

   (b) Module 2
   For Wheel-2:

$$\sum F_x = 0: \ fx_{w2} = 0 \tag{45}$$
$$\sum F_y = 0: \ fy_{w2} + w_w = 0 \tag{46}$$
$$\sum M_{J1} = 0: \ \tau_{w2} = 0. \tag{47}$$

For Trunk-2:

$$\sum F_x = 0: \ fx_{l2} - fx_{l1} + fx_{w2} = 0 \tag{48}$$
$$\sum F_x = 0: \ fy_{l2} + fy_{w2} - fy_{l1} - w_l = 0 \tag{49}$$
$$\sum M_{J1} = 0: \ k_2\phi_2 - k_1\phi_1 + w_l[(l/2)cos\phi_2 - csin\phi_2] - fx_{l1}(l + l_0)sin\phi_2 + fy_{l1}(l + l_0)cos\phi_2 + fx_{w2}lsin\phi_2 - fy_{w2}lcos\phi_2 - \tau_{w2} = 0. \tag{50}$$

   (c) Module 3
   For Wheel-3:

https://doi.org/10.1017/S0263574715000685 Published online by Cambridge University Press

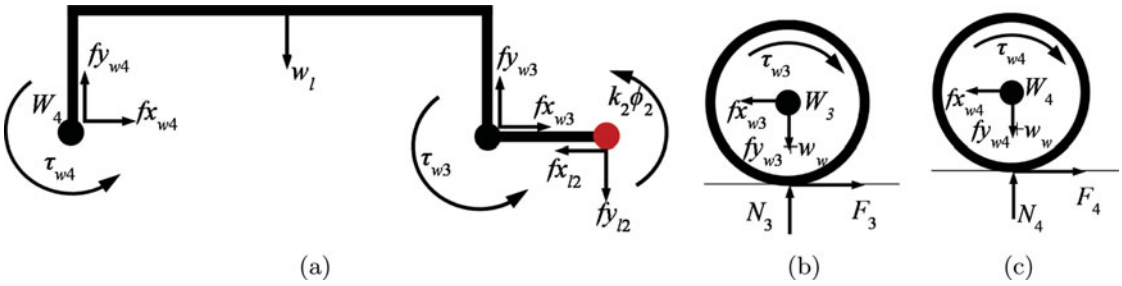


Fig. 20. (a) Trunk 3 (b) Wheel 3 (c) Wheel 4.

$$\sum F_x = 0: \ fx_{w3} - F_3 = 0 \tag{51}$$
$$\sum F_y = 0: \ fy_{w3} - N_3 + w_w = 0 \tag{52}$$
$$\sum M_{J1} = 0: \ \tau_{w3} - F_3 r = 0. \tag{53}$$

For Wheel-4:

$$\sum F_x = 0: \ fx_{w4} - F_4 = 0 \tag{54}$$
$$\sum F_y = 0: \ fy_{w4} - N_4 + w_w = 0 \tag{55}$$
$$\sum M_{J1} = 0: \ \tau_{w4} - F_4 r = 0. \tag{56}$$

For Trunk-3:

$$\sum F_x = 0: \ fx_{l2} - fx_{w3} - fx_{w4} = 0 \tag{57}$$
$$\sum F_y = 0: \ fy_{w4} + fy_{w3} - fy_{l2} - w_l = 0 \tag{58}$$
$$\sum M_{J1} = 0: \ -k_2\phi_2 + w_l(l/2) + fy_{l2}(l + l_0) - fy_{w3} l - \tau_{w3} - \tau_{w4} = 0. \tag{59}$$

(d) Minimal set of equations for phase-2
The minimal set of equations for Phase-2, i.e., Eqs. (6)–(10) are obtained by eliminating the constraint forces (fx_{li}, fy_{li},fx_{wi} and fy_{wi}) from the above equations. The steps are illustrated below:

(i) It can be seen from Eqs. (39), (42), (45), (48), (51) and (54) that $fx_{w1} = N_1$, $fx_{l1} = fx_{w1} = N_1$, $fx_{w2} = 0$, $fx_{l2} = fx_{l1} - fx_{w2} = N_1$, $fx_{w3} = F_3$ and $fx_{w4} = F_4$, respectively. Therefore, Eq. (6) is obtained by substituting the values fx_{l2}, fx_{w3} and fx_{w4} from the above into Eq. (57).

(ii) It can be seen from Eqs. (40), (43), (46), (49), (52) and (55) that $fy_{w1} = w_w - F_1 = 0$, $fy_{l1} = w_l + fy_{w1} = w_l + w_w - F_1$, $fy_{w2} = -w_w$, $fy_{l2} = -fy_{w2} + fy_{l1} + w_l = 2w_w + 2w_l - F_1$, $fy_{w3} = N_3 - w_w$ and $fy_{w4} = N_4 - w_w$, respectively. Hence, Eq. (7) is obtained by substituting the values fy_{w4}, fy_{w3} and fy_{l2} from the above into Eq. (58).

(iii) Equation (8) is obtained by substituting values of fx_{w1}, fy_{w1} and τ_{w1} from Eqs. (39)–(41), respectively, into Eq. (44).

(iv) Equation (9) is obtained by substituting values of fx_{l1}, fy_{l1}, fx_{w2}, fy_{w2} and τ_{w2} from $item - i$ above, $item - ii$ above, Eqs. (45)–(47), respectively, into Eq. (50).

(v) Equation (10) is obtained by substituting values of fy_{l2}, fy_{w3}, τ_{w3} and τ_{w4} from $item - ii$ above, Eqs. (52), (53) and (56), respectively, into Eq. (59).

References

1. D. Apostolopoulos, "Analytic configuration of wheeled robotic locomotion," Carnegie Mellon University (2001).

https://doi.org/10.1017/S0263574715000685 Published online by Cambridge University Press

2. S. Avinash, A. Srivastava, A. Purohit, S. V. Shah and K. M. Krishna, "A Compliant Multi-Module Robot for Climbing Big Step-Like Pbstacles," *IEEE International Conference on Robotics and Automation (ICRA)*, Hong Kong, China (2014).
3. S. Avinash, S. V. Shah and K. Madhava Krishna, "Wheel Torque Optimization for a Compliant Modular Robot," *National Conference on Machines and Mechanisms (NaCoMM)*, Roorkee, India (2013).
4. P. Lamon, A. Krebs, M. Lauria, R. Siegwart and S. Shooter, "Wheel Torque Control for a Rough Terrain Rover," *IEEE International Conference on Robotics and Automation (ICRA)*, New Orleans, USA (2004).
5. P. Lamon and R. Siegwart, "Wheel Torque Control in Rough Terrain-Modeling and Simulation," *IEEE International Conference on Robotics and Automation (ICRA)*, Barcelona, Spain (2005).
6. T. Thueer, P. Lamon, A. Krebs and R. Siegwart, "Crab-exploration Rover with Advanced Obstacle Negotiation Capabilities," *Proceedings of the 9th ESA Workshop on Advanced Space Technologies for Robotics and Automation (ASTRA)*, Noordwijk, The Netherlands (2006).
7. A. Krebs, T. Thueer, E. Carrasco and R. Siegwart, "Towards Torque Control of the Crab Rover," *Proceedings of the 9th International Symposium on Artificial Intelligence, Robotics and Automation in Space (iSAIRAS)*, Los Angeles, USA (2008).
8. A. Krebs, F. Risch, T. Thueer, J. Maye, C. Pradalier and R. Siegwart, "Rover Control Based on an Optimal Torque Distribution-Application to 6 Motorized Wheels Passive Rover," *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Taipei, Taiwan (2010).
9. B. Xu, C. Pradalier, A. Krebs, R. Siegwart and F. Sun, "Composite Control Based on Optimal Torque Control and Adaptive Kriging Control for the Crab Rover," *IEEE International Conference on Robotics and Automation (ICRA)*, Shanghai, China (2011).
10. K. Turker, I. Sharf and M. Trentini, "Step Negotiation with Wheel Traction: A Strategy for a Wheel-Legged Robot," *IEEE International Conference on Robotics and Automation (ICRA)*, St. Paul, USA (2012).
11. T. Estier, Y. Crausaz, B. Merminod, M. Lauria, R. Piguet and R. Siegwart, "An innovative space rover with extended climbing abilities," *Proc. Space Robot.* **2000**, 333–339 (2000).
12. A. Krebs, T. Thueer, S. Michaud and R. Siegwart, "Performance Optimization of All-Terrain Robots: A 2d Quasi-Static Tool," *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Beijing, China (2006).

https://doi.org/10.1017/S0263574715000685 Published online by Cambridge University Press