MASTER

Slip avoidance for wheeled mobile robots using actuation constraints

Huijser, F.C.A.

Award date:
2019

Slip Avoidance for Wheeled Mobile Robots using Actuation Constraints

CST-number: CST2019.041
Author: F.C.A. Huijser, ID-number: 0769026
Coach: dr. C.A. Lopez Martinez
Committee: prof.dr. H.P.J. Bruyninckx, dr.ir. M.J.G. van de Molengraft, dr.ir. I.J.M. Besselink

Abstract — Avoiding excessive wheel slip is an important requirement for the motion control of wheeled mobile robots, owing to concerns regarding steerability, safety and energy efficiency. This work proposes a slip avoidance method for the ropod platform, an eight-wheeled autonomous robot designed in the ROPOD project that is intended to handle logistical tasks in a hospital. As such tasks include the transportation of legacy loads through a dynamic environment, the method must be able to handle changing and effectively unknown wheel slip characteristics. The proposed method aims to constrain each actuator's input to the maximum allowable value before excessive slip occurs, based on the output of a slip detector. Adhering to these constraints, a higher level motion controller distributes a given platform-level control effort over the actuators, downscaling this reference if necessary to keep the distribution problem feasible. By separately updating each actuation constraint, the method enables the motion controller to make use of the robot's actuator redundancy to increase driving performance. Through downscaling the reference control effort, the amount by which wheel slip impairs trajectory following can be communicated to a higher level control loop. The proposed method is implemented and tested on a prototype of the ropod platform. Its performance is evaluated and a number of recommendations for further improvement of the method are made.

Keywords — wheel slip, mobile robot, control allocation

I. Introduction

Over the past few decades, the use of mobile robots has been a major contributor towards increasing labour productivity in many fields of activity. An active area of development is the application of Automated Guided Vehicles (AGVs) for logistical purposes. The ROPOD project [1] aims to develop an AGV that is both widely deployable and cost-efficient, to be used in hospitals for indoor transportation of goods, including existing legacy containers. A key aspect of these AGVs, dubbed "ropods", is their modular design according to the platform concept, originating from the automotive industry [2]. Each ropod is equipped with four swivel casters containing two separately actuated wheels, creating an eight-wheeled platform capable of handling heavy loads. It is important to note the resulting system is over-actuated, possessing eight actuators to control three degrees of freedom.

A significant challenge in controlling the motion of any wheeled vehicle lies in the occurrence of wheel slip. Slip can occur whenever a wheel loses traction with its driving surface. This typically happens when the vehicle undergoes a high acceleration or drives over a slippery surface. A vehicle with one or more slipping wheels experiences a loss of controllability and is at risk of skidding, resulting in undesired movement. Additionally, slipping wheels cannot convert their actuation energy to useful motion as they would under normal rolling conditions, incurring energy loss which is particularly undesired in battery-powered vehicles. The ropod platform is intended to handle loads of varying size and weight in a dynamic environment, including changing floor conditions and obstacles to be avoided. As such, the platform is prone to have to execute manoeuvres that incur slip, such as a high acceleration or braking motion to avoid collision with a moving obstacle. While in operation, unintended motion of the platform due to skidding or even a decrease in controllability is unacceptable in a hospital environment where the safety of humans might be compromised. Being battery-powered, the loss of energy due to slip is undesirable as it shortens the length of time for which the platform can stay in continuous operation. It is therefore an important requirement in the motion control of the ropod platform to both prevent the occurrence of slip as much as possible and ensure a swift recovery whenever slip manifests.

Controlling wheel slip is not an issue limited to the motion control of mobile robots. In the automotive industry, control approaches such as the anti-lock braking system (ABS) and traction control (TC) [3] have been applied for the better part of the past century. Such systems aim to prevent skidding of wheels while achieving high driving performance by controlling wheel torque. For over-actuated electric vehicles such as the ropod platform, the possibility exists to distribute a control effort imposed at the vehicle level over different actuators in order to avoid slip that would occur if a single actuator were to be used. Making use of this opportunity requires the slip controller to be an integral part of the overall motion control of the vehicle instead of being designed as a monolithic system controlling slip for each wheel. The development of such motion control methods is an active area of research for electric vehicles in general. This work explores the possibility to create a slip control method for the ropod platform that makes use of the ropod platform's actuator redundancy wherever this is beneficial.

A. Related work

This subsection gives an overview of existing approaches towards controlling wheel slip and performing control allocation between multiple actuators. Such approaches comprise purely theoretical control laws, control methods tested in simulations and live-running controllers implemented on vehicles.

Wheel slip control

The most well-known approaches towards controlling wheel slip are found in the form of ABS and TC systems, owing to their widespread use in the automotive industry. In recent years, the advent of all-wheel drive electric vehicles has led to the development of new approaches for ABS and TC control systems. Electric motors possess a significantly lower actuation delay compared to conventional systems such as internal combustion engines and friction brakes, allowing for slip control functions to operate at a higher frequency. Additionally, the individual control over in-wheel motors allows for the traction and braking dynamics of a vehicle to be controlled under all driving conditions instead of only under emergency conditions [4]. In general, the aim of such systems is to maintain a certain amount of slip at each wheel, maximizing traction force while preserving vehicle stability. Providing driving comfort is one of their main design goals, as they are developed for cars with human drivers. In other domains, such as motion control for Wheeled Mobile Robots (WMRs), the prevailing approach is to design controllers such that a trajectory following error is minimized.

Control approaches that aim to maintain a certain amount of wheel slip are mostly seen in cars, where control input is provided by a driver and preserving driving comfort forms a requirement on controller behaviour. Such methods require real-time estimates of wheel slip and tyre-road friction, typically involving extensive modelling of vehicle and tyre dynamics. Provided these parameters are known, controlling a target slip can be achieved through PID control per wheel, both for obtaining TC [5] and ABS functionality [6]. Controlling wheel slip at all times instead of only during acceleration and braking manoeuvres can be achieved using a model predictive control structure [7]. However, the stringent requirements on the vehicle parameters that have to be known prevents such methods from leaving simulation environments. Methods to include a comprehensive estimation of these parameters in the controller design have been developed [8], but a major downside is the unpractically large computational effort required to implement these in a live-running fashion. Specifically regarding ABS architectures for car-like vehicles, the possibility exists to optimally divide braking actuation over the electric motors and hydraulic brakes [9]. A different approach towards controlling the slip ratio is to adapt the saturarion limit of the wheel torque while tracking a reference wheel speed, significantly reducing the complexity of the control problem [10] [11]. Again however, these methods rely on extensive dynamic modelling and real-time estimation of tyre-road friction. Though mostly limited to cars, methods purely focusing on controlling optimal wheel slip have been investigated for nonholonomic WMRs [12]. Such methods are limited to simulation environments as significant computational effort is required to compute the control inputs.

Instead of having human drivers dictating control input, WMRs possess a motion control structure computing the required control effort to achieve a desired motion, usually based on modelled vehicle dynamics. Typically, such motion controllers aim to solve a trajectory following problem. The predominate approach towards slip control is to take slip into account in the overall vehicle dynamics considered by the motion controller. This way, slip control is part of the overall motion control problem of the robot. For nonholonomic robots, numerous control methods have been devised to address trajectory following in the presence of both longitudinal slip and skid while turning [13] [14] [15]. These methods employ dynamic models that rely on specific hardware configurations and are typically of such complexity that they cannot leave simulation environments. For robots intended to operate in offroad conditions, estimation methods for the local slip characteristics are of particular interest [16] [17]. As with similar methods developed for cars, the feasibility of such methods to be implemented in a live-running fashion is severely limited by their computational cost. Aside from WMRs, real-time trajectory-tracking controllers have been developed for tracked vehicles serving outdoor agricultural purposes [18] [19]. Again, these methods rely on a very specific hardware configuration.

Control allocation

The motion control structure for a vehicle with redundant actuators commonly includes three levels. At the highest level, the overall motion of the vehicle is controlled through virtual control efforts. Second, a control allocation algorithm manages the output of each actuator in order to deliver the required vehicle-wide control effort. Finally, low level controllers ensure each actuator produces its desired output [20]. In addition to meeting the control effort requirement of a higher level motion controller, a control allocation algorithm can aim to achieve secondary objectives, such as minimizing power consumption or preffering certain specific actuators. The control allocation problem can be formulated as a mixed objective function, combining error minimization with a weighted total control effort minimization objective [21]. For wheeled vehicles, the effort of individual actuators can be weighted to vertical load in order to avoid slip [22]. To determine the proper weighting for each actuator, extensive knowledge of the relevant forces acting on each wheel is required. Typically, more complex approaches focus on the specific configuration and characteristics of the vehicle.

As with wheel slip control systems, electric car-like vehicles with human drivers are a major area of interest regarding the development of control allocation methods. A relatively simple approach is to control the torque distribution ratio between the front and rear wheels to minimize the yaw velocity error and between the left and right wheels to minimize the lateral velocity error [23]. A more complex approach makes use of several levels of control, allowing to switch between control strategies according to circumstances and driver input to optimize for energy consumption [24]. If the exact effects of different torque distributions on drivetrain losses are analytically computed beforehand, a live-running distribution method can be implemented using a lookup table [25]. All of these methods are designed for the drivetrains of car-like vehicles and require precise tuning of control parameters. Regarding control allocation methods that specifically aim to compensate for individual wheel slip, several methods have been developed. These methods aim to minimize the squared sum of the wheel slip ratios [26] or a thereof derived energy loss [27], penalizing slipping wheels accordingly. Again however, these methods are specifically designed for car-like vehicles. Additionally, they rely on an accurate measurement of slip at the wheel level.

B. Problem statement and objective

The ropod platform is intended to autonomously transport a wide variety of loads through a highly dynamic environment. Per task it performs, the dynamics related to the interaction between the robot and its driving surface might change and are effectively unknown. Additionally, unpredictable local floor conditions can occur while the ropod is in motion. Owing to safety and energy efficiency concerns, it is desirable for the ropod's motion control system to compensate for wheel slip. The objective of this work is to develop a slip avoidance method for the ropod platform that complements its existing motion control structure. Considering the ropod's intended use, it is not necessary for this method to maximize driving performance by controlling the exact amount of wheel slip: merely preventing excessive wheel slip is sufficient. Integration in the motion control hierarchy has to be taken into account: the method must not behave as a black-box system for higher levels of motion control and planning but rather communicate clearly what influence it exerts on the overall motion of the ropod. Given the ropod's actuator redundancy, this work will also explore the possibility to increase driving performance under hazardous conditions by including control allocation over the different wheels. Additionally, it is desirable to minimize the method's reliance on a single hardware configuration, e.g. a specific set of actuators and sensors, considering the intended modularity of the ropod's design. In order to run in a real-time fashion, any implementation of the method must be computationally inexpensive. To the best of the author's knowledge, no slip avoidance method exists as of yet that meets all these requirements. As part of this project, the devised method will be implemented on a ropod prototype and evaluated through experiments.

C. Paper organization

First, the approach of the proposed slip avoidance method is detailed in Section II. Section III details how the proposed method is implemented on the ropod platform. In Section IV, the proposed method is experimentally validated and an evaluation of its performance is given. Finally, Section V concludes the paper and gives recommendations for future work.

II. Approach

This section details the proposed methodology towards slip avoidance for the ropod platform. First, the approach towards avoiding slip for a single wheel is derived. From there, the proposed integration of this approach in the ropod's motion control structure is explained.

A. Wheel slip avoidance

The main purpose of the ropod platform is the transportation of goods in a hospital. As such, a ropod will typically be tasked to navigate hallways while hauling a load. Consequently, the ropod's wheels are expected to be at risk of slipping during straight-line motion: either during the execution of acceleration and braking manoeuvres or when a change in local floor conditions is encountered. In this project, only longitudinal slip of the ropod's wheels occurring during straight-line motion is considered. Although lateral slip might occur to some extent during cornering manoeuvres, it is assumed not to reach significant values compared to longitudinal slip under normal operating conditions. For a single wheel, let the longitudinal wheel slip ratio be defined as

$$\lambda_w = -\frac{\dot{y} - r_w \dot{\omega}_w}{\dot{y}}, \tag{1}$$

with $\dot{y}$ denoting the ropod's forward velocity in $\frac{[m]}{[s]}$, $\dot{\omega}_w$ the wheel's rotational velocity in $\frac{[rad]}{[s]}$ and r_w the wheel's radius in $[m]$. According to this definition, wheel slip during acceleration of the ropod is represented with a positive value of λ_w and wheel lock-up during braking with a negative value. Only non-negative velocities are considered. As described in the problem statement, the objective of a prospective slip avoidance method is to prevent the amount of wheel slip, represented by λ_w, from reaching a certain value considered to be excessive. The control objective per wheel is then defined as preventing λ_w from exceeding some predetermined value $\lambda_{w,max}$ representing the lower limit of said excessive slip. For clarity, the remainder of this analysis is conducted for positive values of λ_w: an analogue derivation can be made for negative values of λ_w.

The amount of slip a wheel exhibits is related to the tractive force F_t that enables the wheel to perform a rolling motion over a driving surface. The tractive force is defined as

$$F_t = \mu(\lambda_w) F_N, \tag{2}$$

where μ denotes the surface adhesion coefficient, which is a function of λ_w, and F_N denotes the normal force acting on the wheel. The shape of the non-linear function $\mu(\lambda_w)$ is dependent on the local tractive properties of the driving surface. In Figure 1, an example of typical slip characteristics of car wheels for different road surface conditions is shown.

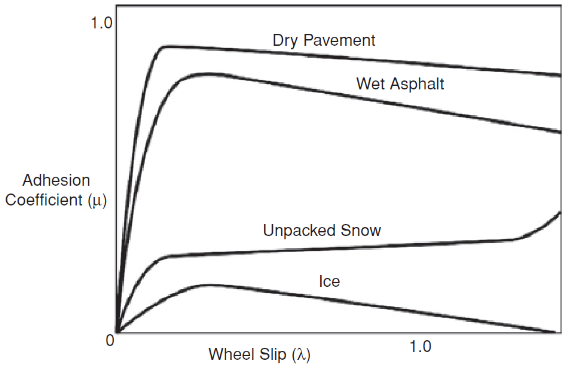


Fig. 1: Typical wheel slip characteristics of car wheels for different road surface conditions [3].

The ropod platform is intended to operate in a highly dynamic environment and as such, required to be able to handle unknown local floor conditions. Consequently, the slip characteristics of the ropod's wheels are considered to be unknown. However, it is assumed the shape of the function $\mu(\lambda_w)$ does not change during the execution of a slip-inducing manoeuvre, such as performing a straight-line acceleration. This way, during said manoeuvre, for the predefined maximum slip ratio of $\lambda_{w,max}$, there exists a corresponding constant value $\mu(\lambda_{w,max}) = C_1$ where the tractive force reaches the maximum allowed value of

$$\begin{aligned} F_{t,max} &= \mu(\lambda_{w,max})F_N \\ &= C_1 F_N. \end{aligned} \tag{3}$$

Note that depending on the shape of $\mu(\lambda_w)$, this value might occur for more than one slip ratio: in this case, $F_{t,max}$ represents the value occurring for the lowest amount of slip. The actuation of the wheel comes in the form of torque applied by the motor axle. As detailed in Appendix A, in order to not exceed $F_{t,max}$, the applied torque τ must not exceed the maximum value of

$$\tau_{max} = P, \tag{4}$$

where P is an unknown variable, assumed constant for the duration of the slip-inducing manoeuvre. The desired slip avoiding behaviour of the wheel is achieved when P is chosen such that it represents the limit for excessive slip and τ is controlled accordingly. Therefore, at the wheel level, the strategy of the proposed slip avoidance method is to permanently limit the torque applied to the wheel to τ_{max} and ensure this value is kept adequately updated to represent the limit for excessive slip.

B. Control allocation

At its core, the ropod platform consists of a rigid frame actuated by four swivel casters which consist of two separately actuated wheels. Consequently, the ropod as a whole forms an over-actuated system which possesses eight actuators to control three degrees of freedom. A schematic representation of a ropod's wheel layout is shown in Figure 2.

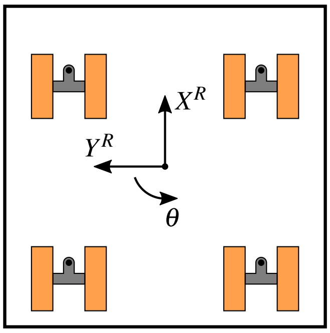


Fig. 2: Schematic representation of the ropod platform's wheel layout. The platform possesses eight wheels to control two translations and one rotation.

The control effort requested from the wheels is the sum of the control effort required to achieve a desired motion of the entire platform and any additional control effort required to prevent undesired rotation of the caster pivots during said motion. As the latter is indispensable to the motion control of the platform, it is not to be altered for the purpose of slip avoidance. As such, only the control effort relating to the motion of the entire platform is of interest. At the platform level, the ropod's motion is controlled through the wrench $\vec{w}_p$ applied on the platform, which comprises a vector of virtual control efforts:

$$\vec{w}_p = [f_{xr}, f_{yr}, \tau_{\theta r}]^T, \tag{5}$$

where f_{xr} and f_{yr} represent the forces applied to the platform in the local coordinates X^R and Y^R, and $\tau_{\theta r}$ represents the torque applied on the platform [28]. The wrench relates to the vector $\vec{\tau}_w$ containing the wheel torques according to

$$\vec{w}_p = G^T S(q)\vec{\tau}_w, \tag{6}$$

where G^T and $S(q)$ are matrices which together map the vector $\vec{\tau}_w$ consisting of the wheel torques to the wrench applied on the platform, making use of the vector q representing the full state of the ropod, as detailed in [28]. The resulting control allocation objective is formulated as the following quadratic optimization problem [28]:

$$\begin{aligned} \underset{\vec{\tau}_w}{minimize} \quad & \vec{\tau}_w^T P_w \vec{\tau}_w + \rho\alpha, \\ s.t. \quad & G^T(q)S(q)\vec{\tau}_w = (1-\alpha)\vec{w}_p, \\ & \vec{\tau}_{min} \le \vec{\tau}_w \le \vec{\tau}_{max}, \\ & 0 \le \alpha \le 1. \end{aligned} \tag{7}$$

The goal of this optimization is to meet the required platform-level control effort $\vec{w}_p$ while minimizing the total control effort in terms of wheel torques $\vec{\tau}_w$. The individual wheel torques in $\vec{\tau}_w$ can be weighted according to the matrix P_w. Additionally, $\vec{\tau}_w$ is constrained according to the values of $\vec{\tau}_{max}$ and $\vec{\tau}_{min}$, providing respectively upper and lower bounds on the individual wheel torques. To ensure the problem is always feasible, $\vec{w}_p$ can be scaled using the auxiliary variable α, which is penalized in the cost function according to the parameter ρ.

In order to avoid excessive slip, the torque applied to each wheel must be limited to the maximum value of τ_{max} at all times. Therefore, the aim of the proposed slip avoidance method is to have the constraints on each element of the wheel torque vector $\vec{\tau}_w$ in the optimization function represent these maximum values for the corresponding wheel. As a result, $\vec{\tau}_{max}$ and $\vec{\tau}_{min}$ represent the torque limits in each direction of motion of the wheels. It is assumed that all times holds that $-\vec{\tau}_{min} = \vec{\tau}_{max}$. If, for whatever reason, these limits are not identical for all wheels, control effort can be distributed in a non-uniform manner over the wheels to meet the desired $\vec{w}_p$. No other restrictions are imposed on the control allocation problem. Indeed, the objective function itself is not influenced by the slip avoidance procedure: the objective is still to meet the platform-level control effort while minimizing the total torque produced at the wheels. If the constraints imposed on $\vec{\tau}_w$ are collectively strict enough, it becomes impossible to solve the allocation problem without scaling the desired $\vec{w}_p$ with a nonzero value of α. If the value of ρ is chosen such that scaling the platform-level control effort is penalized more than increasing the wheel torques, the value of α gives a clear indication on how much the desired motion of the ropod is impacted by the current values of the actuation constraints.

Note that this work does not assert the control allocation problem is best solved using this specific objective function. For the purpose of slip avoidance, only adherence to the torque constraints is required. This particular objective function is employed to showcase the possibility of including additional goals to be accomplished through control allocation. If so desired, individual wheel torques can be weighted using the matrix P_w according to e.g. energy efficiency characteristics of the motors. In the remained of this project, it is assumed slip avoidance is the only additional concern of the ropod's motion control. No individual wheels are penalized using P_w, and ρ is set sufficiently high to ensure $\alpha > 0$ only when the allocation problem would otherwise be infeasible. This way, the value of α can be used to provide a platform-level indication of how much wheel slip interferes with the ropod's desired motion.

C. Resulting strategy

The goal of the proposed slip avoidance method is to constrain the control allocation problem with the wheel slip limits contained in $\vec{\tau}_{max}$ and $\vec{\tau}_{min}$, which at all times relate according to $\vec{\tau}_{min} = -\vec{\tau}_{max}$ with $\vec{\tau}_{max}$ defined positive. Although approximated as constants, these limits are unknown and change for each slip-inducing manoeuvre the ropod initiates. In order to keep them adequately updated, the ropod is required to possess the functionality to detect excessive slip at each of its wheels. The proposed update strategy is as follows. Initially, when the ropod starts operation and no wheel slip has been encountered, the torque constraints for each wheel are set to the saturation limit τ_{sat} of the motor. When excessive slip is detected at a wheel, expected to occur when the ropod initiates a manoeuvre such as straight-line acceleration, the corresponding constraint is reduced (though never reduced below zero) until the wheel no longer slips. Depending on the amount by which said constraint is reduced, conservatism is introduced in the control allocation problem. In order to reduce this conservatism and potentially increase driving performance, it can be attempted to increase the value of the constraint once slip has been sufficiently reduced. However, in doing so the wheel is at risk of entering a state of excessive slip again. As slip avoidance is prioritized over driving performance, this strategy is not explored within the frame of this project. It is proposed the decision to attempt to increase the constraints or reset them to τ_{sat} is left to a higher level of motion control, which can evaluate if the ropod is still at risk of slipping.

The proposed slip avoidance method places a number of requirements on different components of the ropod's motion control system. The method itself operates at the control allocation level, situated between the platform-level control of the ropod's motion and the low-level control of the actuators. First, the method requires the presence of a wheel slip detector at a lower level. The purpose of this detector is to communicate the occurrence of excessive wheel slip to the higher levels of control. At a minimum, this communication takes the form of a boolean per wheel to indicate whether excessive slip is present or not. More extensive information could be used to reduce conservatism on the torque constraints, but is not required for the functioning of the proposed method. Secondly, the presence of a higher level controller is required that decides when the risk of raising the torque constraints is taken. Ideally, this controller is able to assess whether conditions relating to slip have changed sufficiently to warrant this risk, such as caused by, for instance, the ropod entering an area with a different driving surface. In addition, it can take into account the scaling applied by the control allocation algorithm to the desired platform-level control effort and decide whether or not driving performance needs to be improved. Regardless of the exact mode of operation of such a controller, it is proposed that at a minimum, all torque constraints are reset to τ_{sat} whenever the ropod concludes an operation cycle. To summarize, the proposed slip avoidance method receives input from a slip detector on the slip state of each wheel and in turn outputs the control effort scaling factor to higher levels of control.

III. Implementation

In this section, the implementation of the proposed slip avoidance method on a prototype of the ropod platform is detailed. First, an overview of the ropod prototype and its existing motion control system is given. Secondly, the integration of the proposed method in the platform's motion control system is explained. Finally, the implementation of the other motion control components required by the slip avoidance method is discussed.

A. Overview of the ropod platform

The ropod platform prototype is shown in Figure 3. The central aspect of its design is the use of modular wheel units, called "Smart Wheels". The Smart Wheel is developed from the PowerWheel 200, depicted in Figure 4, which was developed in the run-up to the ROPOD project [29]. At its base, a Smart Wheel (SWh) consists of a swivel caster with two wheels separately actuated by embedded brushless dc motors. Through embedded electronics, the SWh handles low-level control of the motors itself, providing the control effort required by higher levels of control. In addition, the SWh features extensive sensing capabilities, enabling real-time monitoring of various aspects of its state. In particular, it is equipped with encoders, gyroscopes and an IMU which allow to measure the individual wheel velocities, pivot velocity and acceleration of the SWh.

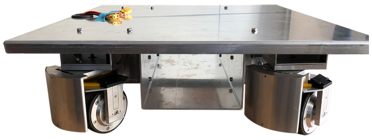


Fig. 3: Prototype of the ropod platform [29].

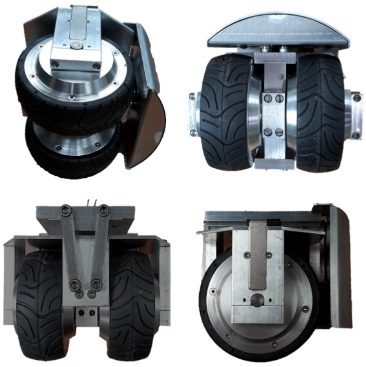


Fig. 4: PowerWheel 200, from which the Smart Wheel is developed [29].

The motion control system of the ropod platform is currently in an active state of development and does not yet possess all the functionality envisioned by the ROPOD project [28]. Within the frame of this work, only the existing lower levels of motion control are considered that allow the ropod to perform the straight-line manoeuvres required to test the proposed slip avoidance method. The lowest level of motion control is handled by the SWhs themselves, which ensure each motor produces the torque as required by the platform-level control system. At the platform-level, the motion control system effectively consists of two control loops operating in parallel. The primary control loop concerns itself with control allocation using the vector of virtual control efforts $\vec{w}_p = [f_{xr}, f_{yr}, \tau_{\theta r}]^T$, as detailed in Section II. Additionally, a second control loop is present for each SWh which damps the velocities of the pivots $\vec{\dot{\delta}}$ in order to remove undesired oscillations that might occur during movement. These controllers consist of a low gain, a lead-lag filter and a low-pass filter. The vector of required wheel torques $\vec{\tau}$ ultimately send to the SWhs is the sum of the output of these two control loops. A consequence of this control structure is that the resulting applied wheel torque does not necessarily adhere to the constraints defined by $\vec{\tau}_{max}$. Therefore, using the proposed slip avoidance method, each constraint has to compensate for the torque applied by the pivot velocity controller in order to avoid excessive wheel slip. As a result, the constraints might deviate from the actual excessive slip limits of the wheels. Controlling the pivot velocity is however an essential part of the motion control system considered in this work: the ropod is not able to perform the intended straight-line acceleration without this control loop. The output of the pivot velocity controller is therefore considered a part of the unknown slip-related dynamics of the ropod and not modified in any way.

As of yet, the ropod's motion control system does not contain a higher level of motion planning capable of providing a reference motion in terms of the wrench $\vec{w}_p$ to the lower levels of control. In this work, a motion control system is used that relies on tracking a given reference velocity $\vec{v}_{sp}$ describing the desired trajectory in terms of the local velocities $\dot{X}^R$, $\dot{Y}^R$ and $\dot{\theta}^R$ of the ropod [28]. The main platform-level control loop contains a controller tasked with following this trajectory, taking the velocity error as input and providing the output $\vec{w}_p$ to the control allocation function. A schematic overview of the ropod's control structure is given in Figure 5. Here, $C_p(s)$ and $C_v(s)$ represent the platform velocity controller and pivot velocity controllers, respectively. The control allocation function $f_\tau(q)$ is implemented according to the approach detailed in Section II. It adheres to the vector of actuation constraints $\vec{\tau}_{max}$, which in turn is determined according to the proposed slip avoidance method. Lower level control and odometry are not depicted separately from the ropod itself.

The aforementioned motion control system is implemented in Simulink, from which it is compiled to an executable to run on the ropod platform. The control loop is set to operate at a frequency of 1000 Hz. As such, any provided velocity reference is sampled at this same frequency. Furthermore, it is assumed the SWhs low-level control system is able to operate within the provided time frame.

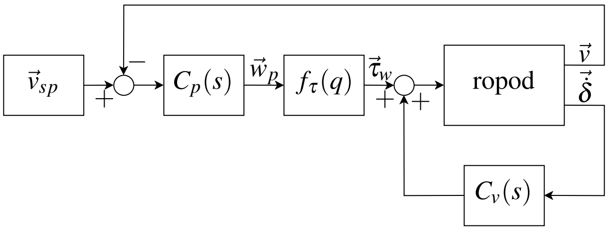


Fig. 5: Schematic overview of the existing platform-level motion control structure, as considered in this work. The proposed slip avoidance method provides the vector of actuation constraints $\vec{\tau}_{max}$ to the control allocation function $f_\tau(q)$.

Algorithm 1 Actuation constraints update algorithm

Inputs:	$\vec{\tau}_{w,k} \in \mathbb{R}$	Applied torque input, [Nm]
	$\vec{s}_k \in \{0,1\}$	Slip detector output, [-]
	$c_k \in \{0,1\}$	Higher level request, [-]
Output:	$\vec{\tau}_{max,k} \in [0, \tau_{sat}]$	Torque constraint, [Nm]
Parameters:	$K \in (0,1)$	Update gain, [-]
	$T_u \in \mathbb{N}^+$	Update wait iterations, [-]
	$\tau_{sat} \in \mathbb{R}^+$	Saturation torque, [Nm]
Variables:	$\vec{t}_{s,k} \in \mathbb{N}$	Update counter, [-]

```
1:  $k = 0$, $\vec{\tau}_{max,0} = \tau_{sat}$, $\vec{t}_{s,0} = 0$
2:  while $k++$ do
3:      $\vec{t}_{s,k} = \vec{t}_{s,k-1} - 1$
4:      for $n = 1 : 8$ do
5:          if $\vec{s}_k(n) = 1$ & $\vec{t}_{s,k}(n) \leq 0$ then
6:              $\vec{\tau}_{max,k}(n) = \mid \vec{\tau}_{w,k-1}(n) \mid -K \mid \vec{\tau}_{w,k-1}(n) \mid$
7:              $\vec{t}_{s,k}(n) = T_u$
8:          else if $c_k = 1$ & $\vec{s}_k(n) = 0$ then
9:              $\vec{\tau}_{max,k}(n) = \tau_{sat}$
10:         else
11:             $\vec{\tau}_{max,k}(n) = \vec{\tau}_{max,k-1}(n)$
```

B. Updating the actuation constraints

The update strategy for the constraints $\vec{\tau}_{max}$ imposed on the control input $\vec{\tau}_w$ proposed in Section II is implemented as part of the ropod's motion control loop. Per motion control iteration, the implemented algorithm is called to update the torque constraint on each wheel based on the input received from the wheel slip detector mentioned in Section II. The minimum required input from the slip detector consists of a boolean indicating whether excessive slip is present or not. Furthermore, the algorithm assumes the presence of a higher level motion planner that decides when to reset the constraints to their initial values. A major additional requirement for the algorithm is to account for the potential difference in the time scales at which the motion control loop operates and on which the slip dynamics take place. Given the control loop's operating frequency of 1000 Hz, it is expected the slip dynamics take place on a larger time scale than the maximum of 1 millisecond available between two control iterations. To account for this larger time scale, the algorithm needs to incorporate the functionality to allow the slip dynamics enough time to resolve after each change in a wheel's constraint. This is achieved by having the algorithm wait a certain number of control iterations after reducing a constraint before it considers doing so again.

A pseudocode description of the implemented algorithm is given in Algorithm 1. The algorithm consists of the following parts:

Line 1: when the ropod starts operation, the initial values of the torque constraints contained in $\vec{\tau}_{max}$ are set to the saturation limit τ_{sat} of the motors. Note that τ_{sat} is assumed equal for all motors in this particular implementation.

Line 2-4: for each iteration k of the motion controller, the counters in $\vec{t}_{s,k}$ are updated and the algorithm loops over all eight wheels.

Line 5-7: if excessive slip is detected at wheel n and the corresponding counter has run its course, $\vec{\tau}_{max}(n)$ is reduced. The input accepted from the slip detector consists of a binary value, indicating whether excessive slip is present or not. Given the previous input $\vec{\tau}_{w,k-1}(n)$ to the wheel, $\vec{\tau}_{max}(n)$ is set to a value below this input, determined by the gain K. Whenever the value of $\vec{\tau}_{max}(n)$ is reduced in this manner, the corresponding counter is set to $\vec{t}_s(n) = T_u$, prompting the algorithm to wait for at least T_u iterations before reducing $\vec{\tau}_{max}(n)$ again.

Line 8-9: if no slip is detected and a higher level motion controller requests it, the actuation constraints are reset to τ_{sat}. This can happen, for instance, when the ropod concludes a slip-inducing manoeuvre or enters an area with a different driving surface. As with the input from the slip detector, this input takes the form of a binary value.

Line 10-11: when no slip is detected and no request from a higher level controller is received, the current value for the constraint is maintained.

Of the three adjustable parameters in the algorithm, only the saturation torque τ_{sat} is a fixed property of the motors. Both the gain K and number of wait iterations T_u are tunable: their values impact the overall performance of the algorithm. The value of K determines the size of the step by which τ_{max} is reduced for a wheel. A large value of K results in a large reduction of τ_{max} and vice versa. In situations where multiple reduction steps are required to reach the desired constraint value, choosing a larger value for K might reduce the number of steps and therefore the total time required to recover from excessive slip. For cases where a single reduction step is already sufficient, a smaller value for K might reduce conservatism in the resulting constraint. As the ropod is expected to face a host of different local driving conditions, K will have to be chosen such that it represents a trade-off between recovery time and constraint conservatism. The value of T_u determines how many controller iterations lie between two subsequent reduction steps. It is assumed a minimum value for T_u exists below which the algorithm does not allow for enough time to pass for the wheel to recover from excessive slip, even if τ_{max} is sufficiently reduced. Furthermore, there might be a relation between the recovery time and conservatism in the updated constraint. If the updated value of τ_{max} lies well beneath the physical slip limit, the wheel might need less time to recover from excessive slip compared to when τ_{max} lies close to the slip limit. An obvious downside to choosing a large value for T_u is an increase in recovery time if multiple reduction steps are to be performed. As a swift recovery from excessive slip is prioritized over driving performance, T_u will be chosen such that recovery time is minimized. In Section IV, the effect of choosing different values for both T_u and K on the algorithm's performance is explored through experiments.

In addition to the desired constraints imposed by $\vec{\tau}_{max}$, the control allocation optimization described by equation 7 requires an adequate value for the parameter ρ penalizing scaling of the reference virtual control effort. At all times, this value needs to be sufficiently high so that scaling the reference is penalized more than exerting control effort at the wheels.In order to achieve this, α is penalized according to

$$\rho_k = N * \max(\vec{\tau}_{max,k})^2, \tag{8}$$

with $N = 8$ the number of wheels of the ropod platform. The torque applied to the wheels is penalized in the objective function according to $\vec{\tau}_w^T P_w \vec{\tau}_w$, with the weighting matrix P_w kept as the identity in this work. As such, it is always less expensive to increase the applied torque than it is to scale the reference. This way, scaling is only performed when the desired control effort of the ropod can not be delivered.

The algorithm described in this section operates under the assumption that the measured excessive slip at a wheel is the direct result of the torque applied to the wheel during the previous iteration of the motion control loop. In reality, due to the assumed larger time scale at which the slip dynamics take place, said slip is the result of the torque applied over a longer period of time, i.e. over multiple previous iterations. To take this into account, the torque input $\vec{\tau}_{w,k-1}$ considered in Algorithm 1 would have to be modified to represent the applied torque on a larger time scale. This can for instance be achieved by applying a low-pass filter over the applied torque before making use of it in the constraint update algorithm. For the straight-line manoeuvres considered in this work, it is assumed the torque applied to each wheel does not change significantly for at least the time it takes for excessive slip to occur. Therefore, only the torque applied to the wheels in the previous control iteration is considered. Within in the frame of this project, the effect of this assumption on the algorithm's performance is not investigated. It is understood that this assumption does not hold for more complex manoeuvres involving high frequent changes in the torque applied to the wheels.

C. Other control components

As described in Section II, the slip avoidance method communicates with a slip detector and the higher platform-level motion control of the ropod. As such, it places certain requirements on these components of the ropod's control system. The following describes how these components are implemented for the purpose of testing the proposed slip avoidance method.

Slip detection

The SWhs possess extensive motion sensing capabilities capable of providing odometry data to higher levels of control. Previous work performed as part of the ROPOD project has resulted in the development of a Kalman filter framework which fuses the sensor data from the SWhs to estimate the ropod's state [30]. Additionally, this work proposes a method to detect wheel excessive wheel slip based on the Kalman filter's output. More specifically, it makes use of the residual of the a priori estimation of the wheel's velocity, defining the detection of excessive slip as this residual exceeding the value of 0.15 $\frac{m}{s}$. In this work, the aforementioned method is implemented in the ropod's motion control system to provide the required slip detection capabilities. As per its minimum required functionality, the detector only outputs whether this excessive slip is present or not: no further information on wheel slip is assumed to be available. In future work, the proposed slip avoidance method could be improved by developing a slip detector that can quantify the amount of excessive wheel slip. The constraint reduction gain K could be made dependent on this amount, improving the accuracy of the resulting $\vec{\tau}_{max}$. A simple way to achieve this is to implement a lookup table for different values of K depending on the amount of slip. The slip detector input of Algorithm 1 can readily be changed from a binary value to an integer representing indices for this table. Within the frame of this project however, no further research towards improving wheel slip detection is performed.

Higher level control

The platform-level motion control requires the reference velocity $\vec{v}_{sp}$ as input from a higher level motion planner. In addition, the proposed slip avoidance method relies on input from a higher level to decide, at the bare minimum, when to reset the actuation constraints. As of yet, no high-level planner has been developed for the ropod platform that meets these requirements. For the experiments performed in this work, the ropod is provided a priori with a full reference trajectory including the input c_k for actuation constraints update algorithm, sampled at the operating frequency of 1000 Hz. In accordance with the described minimum requirements, this input consist of a binary value, indicating at which time instant the constraints in $\vec{\tau}_{max}$ have to be reset to τ_{sat}. For the straight-line acceleration manoeuvres considered in this work, this is set to be the case each the ropod concludes an acceleration. In order to include further functionality, this input could be changed to an integer in the same manner as the input from the slip detector. Instead of only being able to reset the constraints, Algorithm 1 could be expanded to include the possibility to piecewise increase the constraint values instead according to

$$\vec{\tau}_{max,k}(n) = \vec{\tau}_{max,k-1}(n) + K_2 \vec{\tau}_{max,k-1}(n) \tag{9}$$

for a certain tunable gain K_2. By increasing $\vec{\tau}_{max}$ in this manner, conservatism in the constraints might be reduced in the face of a slip hazard. Note however that this procedure requires a certain wait time between steps in the same manner as when reducing $\vec{\tau}_{\max}$. As with the gain K, different values for K_2 could be stored in a lookup table. Again however, the possibility of expanding the functionality of higher level motion planning is not explored within the frame of this project.

IV. Experimental results

This section details the experiments performed to evaluate the performance of the proposed slip avoidance method, implemented as described in Section III. First, a suitable value for the number of iterations between update steps T_u is determined. Secondly, the effect of different values for the gain K on the controller's behaviour is investigated. Lastly, the overall performance of the proposed method is evaluated and points for improvement are identified.

A. Tuning of wait parameter T_u

As described in Section III, it is desirable to choose T_u such that the resulting slip recovery time is small. To achieve this, T_u should lie close to the minimum time a wheel needs to recover from excessive slip. To experimentally determine T_u, Algorithm 1 is modified to reduce each constraint in a single step with a fixed amount when slip is detected at the corresponding wheel. By subsequently measuring the number of control iterations it takes for the slip to disappear, a suitable value for T_u can be determined. As it is unknown beforehand by what fixed amount the constraints should be reduced, multiple experiments have to be performed for different step sizes in the range of $(0, \tau_{sat}]$. Additionally, since the slip-related dynamics are dependent on the local driving surface, it is desirable to perform the same experiments on different floor types.

In order to induce excessive slip, a velocity profile corresponding to an acceleration of 4 $\frac{\text{m}}{\text{s}^2}$ is imposed on the ropod for 0.5 s. The initial value for the constraints is set to $\tau_{sat} = 2.8$ Nm. In a series of experiments, these constraints are reduced with a single step over the range of 0.4:0.4:2.8 Nm. Note that the step size of 2.8 Nm corresponds to a constraint reduction to 0. The experiments are repeated for three different floor types: the carpet of the RoboCup field, a plastic laminated floor and a floor consisting of smooth tiles, all of which are present at Gemini-Noord, TU/e. Within this work, these floor types are referred to as “field”, “floor” and “tiles”, respectively. The tractive properties of these surfaces are unknown but assumed different: it is expected the field offers the most traction and therefore leads to the least amount of slip, followed by the floor. Each series of experiments, i.e. each set of experiments over the given range of fixed step sizes for a certain floor type, is performed twice. Between series, the ropod is subjected to a complete power cycle. Per step size and floor type, the number of control iterations it takes for excessive slip to disappear is averaged over all wheels that experienced excessive slip. The results are shown in Figure 6. In Appendix B, a detailed explanation is provided on how the relevant data is extracted from the measurements at the individual wheels.

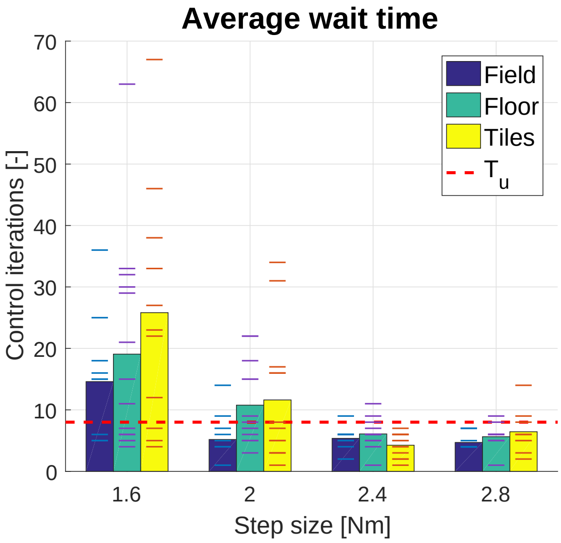


Fig. 6: Average number of control iterations it takes to recover from excessive slip, averaged over all slipping wheels. Additionally, the individual results are depicted. The value chosen for T_u is 8 iterations, corresponding to 8 ms.

In addition to the average number of control iterations, Figure 6 shows the individual result for each wheel, providing an indication of the spread of these results. Only the step sizes of 1.6 Nm and higher prove to be sufficient to recover from excessive slip. As can be seen, there exists a correlation between the step size and the resulting average wait time. This indicates a relation exists between the time required to recover from slip and the conservatism on the updated constraint: the closer the updated constraint lies to the physical slip limit, the longer it takes for excessive slip to disappear. Furthermore, the results show a large spread in the wait times of the individual wheels, indicating a significant difference in the slip characteristics of individual wheels. This is most likely caused by differences in the normal force F_N acting on each wheel. As described in Appendix A, F_N is expected to be dependent on the position of the wheel relative to the motion of the ropod. The consequences of this effect are explained in more detail in the next subsection. The value chosen for T_u is 8 iterations, corresponding to 8 ms given the controller's operating frequency of 1000 Hz. This value is used for all experiments in the remainder of this work.

B. Tuning of gain K

The gain K determines the size of each constraint reduction step. With the wait time T_u fixed, K is the sole adjustable parameter with which the controller's performance can be influenced. As described in Section III, choosing a certain value of K is expected to result in a trade-off between the slip recovery time and conservatism on the reduced constraint of a wheel that experiences excessive slip. In order to evaluate the performance of the controller, this trade-off has to be quantified at the platform level instead of at individual wheels. A measure for the slip recovery time at the platform level can be readily obtained by adding the times spent slipping of the separate wheels together. A measure for the overall conservatism on the actuation constraints is found in the scaling $1-\alpha$ applied by the control allocation algorithm on the desired platform-level control effort. For different values of K, the resulting controller's performance can be expressed by evaluating these two properties.

In a series of experiments, the platform-level scaling and total time spent slipping are measured for values of K in the range 0.1:0.1:0.9. The same velocity reference corresponding to an acceleration of 4 $\frac{\text{m}}{\text{s}^2}$ for 0.5 s is imposed on the ropod. The full constraints update algorithm described by Algorithm 1 is implemented, using a fixed value of 8 iterations for T_u. The experiments are performed for the three previously described floor types. Again, each series of experiments is performed twice. The results are shown in Figure 7 and 8 for the scaling and total slip time, respectively. An example of the measurements at the individual wheels is given in Appendix B.

The results of Figure 7 and 8 are the average of the two performed measurement series. The results of the individual series are depicted as well. As expected, the results follow the trend that an increase of the value of K leads to more scaling applied to the input but less total time spent slipping. The field is shown to indeed offer the most traction out of the three floor types, as both the least amount of scaling is applied here and the least time is spent slipping. The tiles are shown to provide the least amount of traction. In a general sense, no optimal value exists for K. If minimizing input scaling is prioritized, the smallest value for K provides the best performance. A gain of $K = 0.6$ is shown to be a good option if the total slip time is desired to be small. Gains higher than this value lead to unnecessary constraint conservatism on the considered floor types, as they increase scaling without further reducing slip time.

In addition to the direct effect the specific value of K has on the slip recovery of an individual wheel, it also influences platform-level performance in an indirect manner. When a wheel's actuation constraint is reduced due to excessive slip, the torque applied to one or more other wheels will inevitably be increased if the same total control effort is to be achieved. Provided the applied torque on these wheels initially lay below the respective wheel's slip limit, this increase in torque puts these wheels at risk of entering a state of excessive slip as well. Depending on the value of K, the actuation constraints of these wheels might then be reduced to below the initial torque that did not cause slip. This unnecessarily increases both the control effort scaling and total slip time. For the straight-line acceleration considered in this work, this effect might manifests itself between the front and rear wheels of the ropod. As described in Appendix A, the normal force F_N acting on the wheels is expected to be smaller for the front wheels of the ropod. This gives rise to a pattern where the front wheels of the ropod exhibit excessive slip first, followed by the rear wheels after the applied torque to these wheels is increased. For the performed experiments, this pattern can indeed be observed from the individual wheel measurements, as shown in Appendix B. In certain cases, the decrease in performance due to this effect might outweigh the potential benefits from the control allocation the proposed method aims to facilitate. For an increasing value of K for instance, the actuation constraint reduction on the front wheels increases which in turns leads to more torque being applied to the rear wheels. As a result, the rear wheels have an increasing chance to enter a state of excessive slip, upon which their actuation constraints are severely reduced as well. For a sufficiently large value of K, the proposed method might perform poorer than a method that does not attempt to make use of the ropod's actuator redundancy. To evaluate the merits of the proposed method to update all actuation constraints separately, its performance is compared to that of a simplified procedure where all constraints are equally reduced when a wheel exhibits excessive slip.

C. Evaluation of controller performance

To evaluate the merits of the proposed slip avoidance method, its performance is compared to a simplified method that does not take into account the actuator redundancy of the ropod. To this end, Algorithm 1 is modified to no separately update each actuation constraint. Instead, whenever excessive slip is detected at a wheel, all constraints are reduced to the same value as the constraint of the concerned wheel. After each update, the algorithm is set to wait for T_u iterations before reducing the constraints again. Using this implementation, the same experiments for values of K in the range of 0.1:0.1:0.9 are performed. Again, the platform-level scaling and total time spent slipping are measured for the three different floor types, with each series of experiments repeated twice. It is expected this modified method prevents the cascade of excessive slip occurrences at the rear wheels, leading to less time spent slipping and potentially to less control effort scaling. The results are shown in Figure 9, separated per floor type.

Again, Figure 9 shows both the average of the two measurement series and their individual results. As expected, it is shown that the total slip time is significantly lower for the simplified method. In particular, a dramatically reduced slip time can be observed for the tiles. Regarding the scaling, the proposed method is shown to offer better driving performance than the simplified method. Only for the largest values of K, which have already been found to lead to unnecessary constraint conservatism, does the simplified method lead to less scaling. It can therefore be concluded that it does indeed offer an advantage in terms of driving performance to make use of the redundancy in the ropod's actuators when avoiding excessive slip. A required condition is that K is chosen not too large to offset the benefits of being

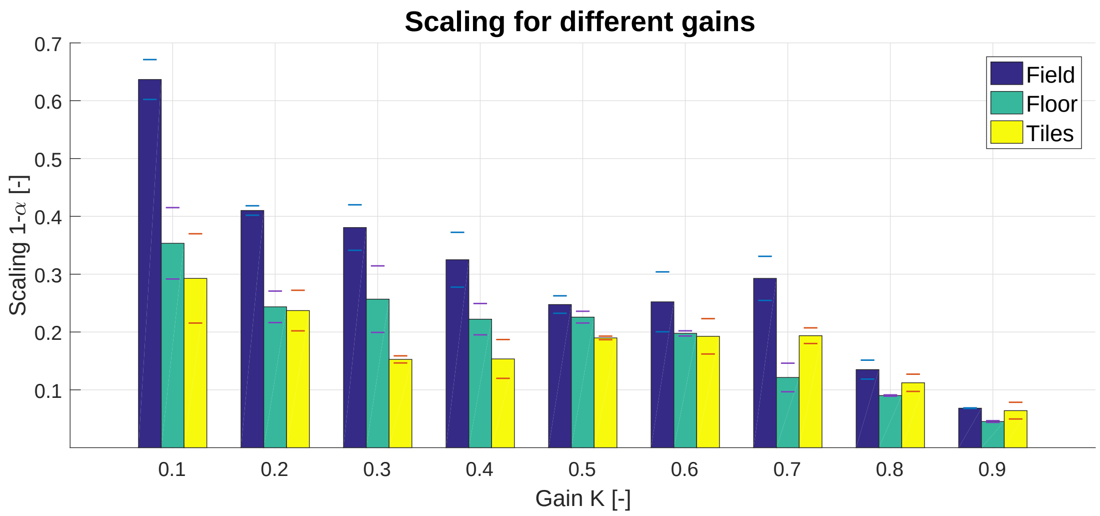


Fig. 7: Platform-level control effort scaling for different values of the gain K. A clear trend is visible where larger values of K lead to more scaling. The results shown are the average of two measurement series.

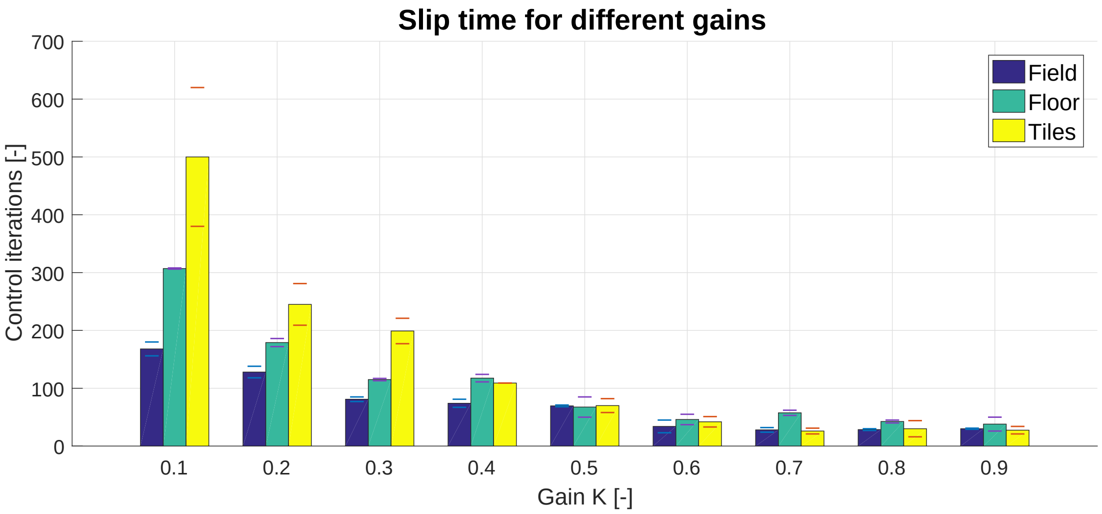
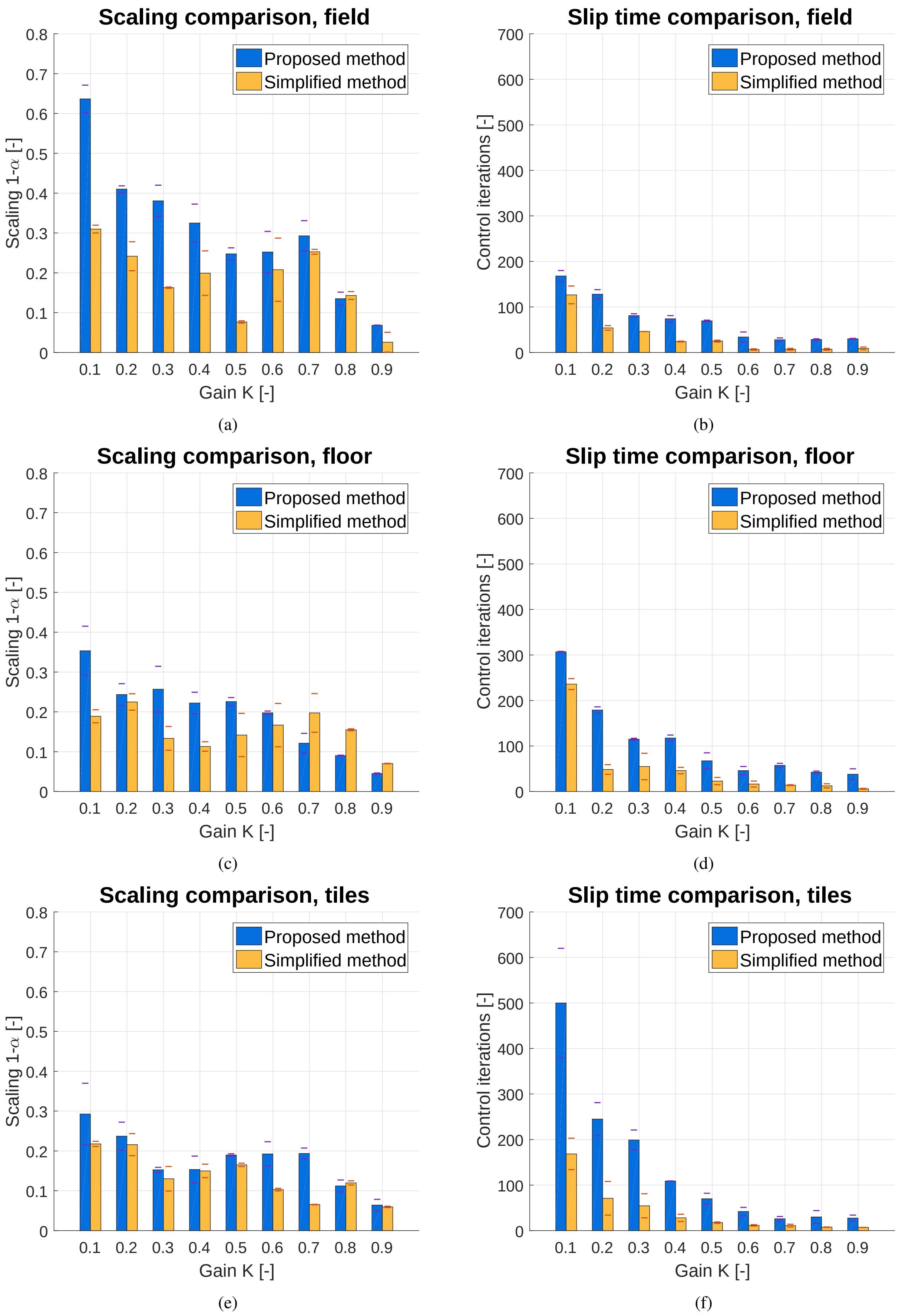


Fig. 8: Slip time summed over all wheels for different values of the gain K. For smaller values of K more time is spent slipping. The result shown are the average of two measurement series.

able to distribute input in a non-uniform manner. Additionally, a number of modifications could be made to the proposed method to account for the aforementioned effect. As proposed in Section III, a possible improvement of the proposed method is to make K dependent on the amount of slip a wheel experiences. This way, the rear wheels are less at risk of experiencing unnecessary conservatism in their actuation constraints when they exhibit excessive slip indirectly induced by the front wheels. Alternatively, if the distribution of F_N between the front and rear wheels can be determined, the ratio between the actuation constraints of the front and rear wheels could be fixed according to this distribution. Whenever the front wheels start exhibiting excessive slip, the actuation constraints of the rear wheels can already be reduced accordingly.

V. Conclusion

In this work, a wheel slip avoidance method for the ropod platform is proposed. Owing to safety and energy efficiency concerns, the aim of the method is to avoid the occurrence of excessive slip at any of the ropod's wheels. This is achieved by constraining the input to each of the ropod's actuators to the maximum allowable value before the corresponding wheel exhibits excessive slip. Adhering to these constraints, the existing higher level motion controller allocates a desired platform-level control effort over the wheels. If necessary in order to keep the control allocation problem feasible, this controller downscales the desired effort. As such, the slip avoidance method complements the existing motion control structure and does not behave like a black-box system to higher levels of control. The actuation constraints are updated based on the output of a slip detector, which



(a)

(b)

(c)

(d)

(e)

(f)

Fig. 9: Comparison between the two actuation constraint update methods in terms of control effort scaling and slip time summed over all wheels. The results are separated per floor type. The results shown are the average of two measurement series.

communicates whether or not a wheel has entered a state of excessive slip. The proposed method is computationally inexpensive and its operating principle is not limited to any single hardware configuration. In its basic form, the method does not require real-time knowledge about the ropod's dynamics and local floor conditions aside from the output of the slip detector. The proposed method has been implemented on a prototype of the ropod platform and experimentally evaluated for straight-line acceleration of the ropod.

The performance of the implemented actuation constraint update algorithm is dependent on the value of two parameters. The first is the minimum number of control iterations T_u to wait between to successive constraint updates: a suitable value for this parameter has been experimentally determined. The second parameters is the gain K, determining the amount by which each constraint is reduced to below the previous actuator input. The effect of the value of K on the algorithm's performance has been investigated through experiments. Using a small value for K results in less scaling being applied the desired control effort, but more total time spent slipping by the wheels. A high value for K leads to less time spent slipping, but more scaling being applied to the control effort. Depending on the desired behaviour of the algorithm, the value of K has to be chosen such that it represents a suitable trade-off between these two performance aspects. Whenever an actuation constraint is reduced, the actuator redundancy of the ropod allows to increase the torque applied to wheels that do not exhibit excessive slip in order to meet the desired total control effort. An unintended effect of this behaviour is that these wheels might enter a state of excessive slip as a direct result of this increase in applied torque, unnecessarily compromising driving performance. The proposed method has been compared to a simplified method that uniformly reduces all constraints whenever excessive slip is detected at a single wheel. This simplified method is shown to reduce the total time spent slipping, but also to increase the scaling applied to the desired control effort. It is therefore concluded the actuator redundancy of the ropod can indeed be used to increase driving performance.

In future work, the proposed slip avoidance method can be expanded in a number of ways. The performance of the method can be readily improved by making use of a more advanced slip detector. In this work, a binary input from the slip detector is considered, providing only the information whether or not a wheel is exhibiting excessive slip. By including more information on the actual amount of slip, the resulting reduced constraint can be set closer to its desired value. This can be achieved by making the value of K dependent on the amount of slip. Another way to reduce conservatism on the constraints is to improve the capabilities of the ropod's higher level motion control system. After a wheel has recovered from excessive slip, a higher level controller can decide to raise the corresponding constraint again with a certain value. Taking this action comes at the risk of the wheel entering a state of excessive slip again. The observed detrimental effect the proposed method has on driving performance in cases where the wheels have different slip characteristics can be counteracted by incorporating knowledge about these characteristics in the constraint update algorithm. For instance, if the ratio between the normal forces acting on the wheels is known, the corresponding constraints can all be updated according to this ratio whenever one of the wheels exhibits excessive slip. Finally, it should be noted the proposed method operates under the assumption that any detected excessive slip is the result of the torque applied to the wheel in the previous control iteration. This assumption might not always hold under normal operating conditions and could have a negative impact driving performance. To account for this, the control inputs considered in the update constraint algorithm can be low-pass filtered to represent the torque applied to each wheel over a longer period of time.

Appendix A
Derivation of wheel dynamics

This appendix details the derivation of the dynamic model of a single wheel as used in Section II. When not experiencing excessive slip, it is assumed the behaviour of the wheel can be simplified to that of ideal rolling. Furthermore, the wheel's stiffness is assumed to be large enough to neglect deformations of the wheel. The dynamic model of a single wheel is shown in Figure A-1. Note that the gravitational and normal force acting on the wheel, opposite and equal in size, are omitted from the diagram for clarity purposes. In Table A-1, an overview is given of the variables used in the dynamic model of the wheel. All modelled dynamic properties of the wheel include, where applicable, the relevant dynamic properties of unmodelled components. For instance, the inertia and viscous friction experienced by the wheel effectively include that of the motor and driving axle.

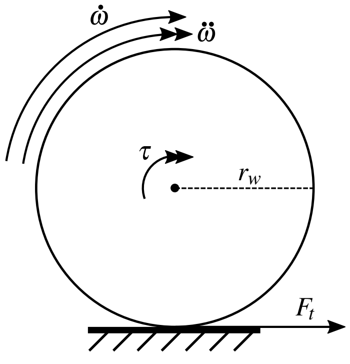


Fig. A-1: Dynamic model of an actuated wheel under ideal rolling conditions. For clarity, the gravitational and normal force acting on the wheel are omitted.

Table A-1: List of variables used in the dynamic model of the wheel.

Var.	Description	Unit
J	wheel inertia	[kg][m]^2
b	viscous friction coefficient	[N][m][s]
τ	applied torque	[N][m]
$\ddot{\omega}$	rotational acceleration	[rad]/[s]^2
$\dot{\omega}$	rotational velocity	[rad]/[s]
r	wheel radius	[m]
F_t	tractive force	[N]
m	total mass	[kg]

The dynamic equation of the wheel is given by

$$J\ddot{\omega} = \tau - b\dot{\omega} - F_t r. \tag{A-1}$$

Rewriting in terms of vehicle velocity, i.e. $\dot{\omega} = \frac{\dot{y}}{r}$ and $\ddot{\omega} = \frac{\ddot{y}}{r}$ gives

$$\frac{J}{r}\ddot{y} = \tau - \frac{b}{r}\dot{y} - rF_t. \tag{A-2}$$

Substituting $\ddot{y} = \frac{F_t}{m}$ and rewriting for τ results in

$$\tau = \frac{F_t(r^2m+J)+mb\dot{y}}{mr}. \tag{A-3}$$

Excessive wheel slip is assumed to occur when the applied torque exceeds a certain value τ_{max} where the traction force reaches the value $F_{t,max}$:

$$\tau_{max} = \frac{F_{t,max}(r^2m+J)+mb\dot{y}}{mr}. \tag{A-4}$$

As shown in Section II, $F_{t,max} = C_1F_N$ for some unknown but constant value of C_1 depending on the local floor conditions. The physical properties of the wheel are effectively unknown, as the system dynamics change per task the ropod executes. However, they are assumed constant for the duration of each task and can therefore be substituted for the unknown constants $C_2 = r^2m+J$, $C_3 = mb$ and $C_4 = mr$. Substituting for $F_{t,max}$, C_2,C_3 and C_4 results in

$$\tau_{max} = \frac{C_1F_NC_2+C_3\dot{y}}{C_4}. \tag{A-5}$$

The size of F_N is understood to be dependent on the type of motion the ropod performs. For instance, during acceleration, the ropod is expected to tilt slightly backwards, resulting in a larger F_N on the rear wheels compared to the front wheels (relative to the direction of motion). However, it is assumed changes in F_N occurring during the execution of a slip-inducing manoeuvre can be neglected. This way, for the duration of said manoeuvre, F_N is considered constant, albeit unknown. This allows for the substitution of the unknown constants $P_1 = \frac{C_1F_NC_2}{C_4}$ and $P_2 = \frac{C_3}{C_4}$ in the dynamic model:

$$\tau_{max} = P_1 + P_2\dot{y}. \tag{A-6}$$

Note that the parameter P_2 represents the viscous friction b experienced by the wheel. As a final simplification, it is assumed the effect of viscous friction can be neglected compared to the other dynamic properties of the model. The possible negative consequence of this assumption is the introduction of conservatism in the model, as the resulting value of τ_{max} will be lower than necessary. Finally, neglecting viscous friction reduces the model to a single unknown parameter:

$$\tau_{max} = P. \tag{A-7}$$

Thus, for each manoeuvre the ropod performs, excessive slip is avoided by choosing the parameter P appropriately and ensuring τ does not exceed τ_{max}.

Appendix B
Explanation of experimental results

This appendix expands on how the experimental results shown in Section IV are obtained from the measurements of the individual wheels. The first experiments shown in Section IV are performed to determine a suitable value for the number of iterations between update steps T_u. Figure A-2 and A-3 show examples of measurement data obtained from a front wheel and rear wheel of the ropod, respectively. The total torque applied on the wheel is the sum of the outputs of the platform-level controller and the pivot velocity controller, as explained in Section III. The slip detector outputs a binary value on the state of the wheel regarding excessive slip. Initially, the platform-level input stays well below the maximum value of $\tau_{max} = \tau_{sat}$. When slip is detected, the constraint is reduced with a fixed value of 2.4 Nm, and the platform-level input lowered accordingly.

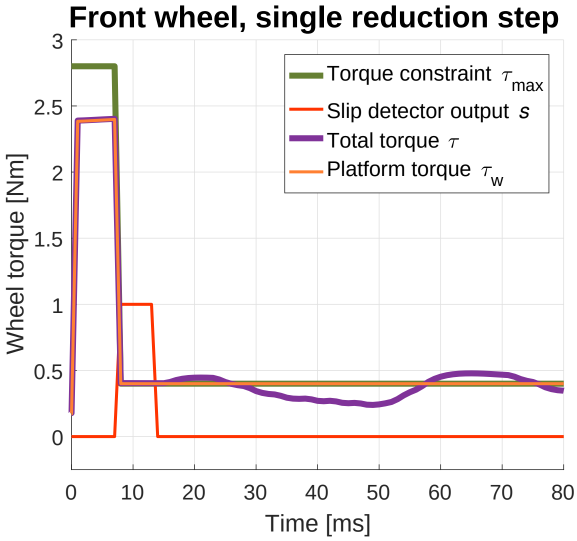


Fig. A-2: Sample measurement data from a front wheel of the ropod during straight-line acceleration. The constraint τ_{max} is reduced from the saturation limit $\tau_{sat} = 2.8$ Nm to 0.4 Nm when excessive wheel slip is detected.

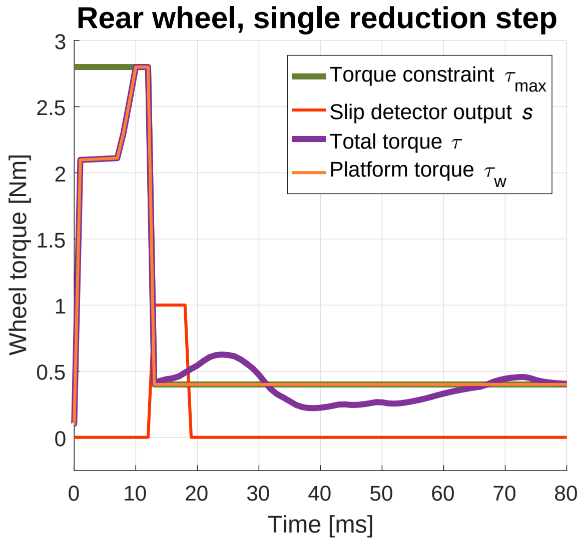


Fig. A-3: Sample measurement data from a rear wheel of the ropod during straight-line acceleration. Excessive slip is detected only after the platform-level torque τ_w is increased.

As shown, it takes a certain amount of time for each wheel to recover from excessive slip after its corresponding constraint is reduced. The relevant information to extract from these measurements is the number of control iterations it takes for slip to disappear after the constraint has been reduced. By averaging this amount over all wheels that exhibit excessive slip during the experiment, the results shown in Section IV are obtained. The difference between the platform-level input and the total torque applied to the wheel is due to the input from the pivot velocity controller. As can be seen, this input can reach significant values in the same order of magnitude as the platform-level input. As the input from this controller is essential for the ropod to perform the intended straight-line motion and is therefore not tampered with.

In Figure A-4 and A-5, examples of wheel measurement data for the full implementation of the proposed method is shown. Again, the figures show data from both a front wheel and a rear wheel of the ropod. The shown measurements are obtained using parameter values of $T_u = 8$ and $K = 0.3$. When slip is detected, τ_{max} is updated to a value of $\tau_w - K\tau_w$. The controller's overall performance is expressed in terms of the scaling applied to the desired total control effort and the total time spent excessively slipping. The latter is obtained by simply adding together the number of control iterations excessive slip is detected at each wheel. The total scaling is evaluated at the end of the ropod's acceleration manoeuvre and is not depicted in the given figures.

As can be seen in both sets of figures, the front wheels consistently exhibit excessive slip ahead of the rear wheels. This is thought to be caused by a difference in the normal force acting on the wheels, brought about by the wheels' positions relative to the direction of motion of the ropod. As can be observed, excessive slip is detected at the rear wheels only after the applied torque to these wheels is increased as an

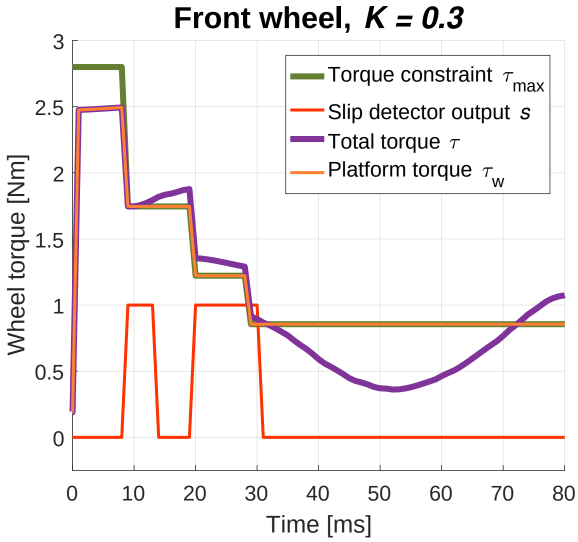


Fig. A-4: Sample measurement data from a front wheel of the ropod using the full implementation of the proposed slip avoidance method. Note the relatively large influence of the pivot controller on the total torque τ applied to the wheel.

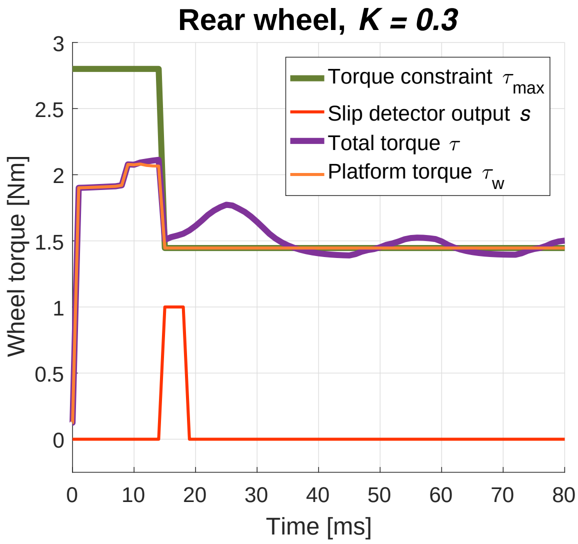


Fig. A-5: Sample measurement data from a rear wheel of the ropod using the full implementation of the proposed slip avoidance method. Excessive slip is detected only after the platform-level torque τ_w is increased.

indirect result of the excessive slip at the front wheels. It is possible the rear wheels would not have started excessively slipping if they had kept their initial applied torque. The negative implications of this phenomenon regarding driving performance warrants further investigation in the merits of the proposed method. Additionally, the input from the pivot velocity controller is again shown to be able to reach a significant value with respect to the total applied torque. This might lead to an increase in conservatism on the resulting constraint.

References

[1] European Commission. Ultra-flat, ultra-flexible, cost-effective robotic pods for handling legacy in logistics. `https://cordis.europa.eu/project/rcn/206247_en.html`. Accessed: 14-03-2019.

[2] ROPOD project: Our Approach. `http://www.ropod.org/our_approach.html`. Accessed: 06-06-2018.

[3] A. Galip Ulsoy, Huei Peng, and Melih Çakmakci. *Automotive Control Systems*. Cambridge University Press, 2012.

[4] Valentin Ivanov, Dzmitry Savitski, and Barys Shyrokau. A survey of traction control and antilock braking systems of full electric vehicles with individually controlled electric motors. *IEEE Transactions on Vehicular Technology*, 64(9):3878–3896, 2015.

[5] Kiumars Jalali, Kai Bode, Steve Lambert, and John McPhee. Design of an advanced traction controller for an electric vehicle equipped with four direct driven in-wheel motors. *SAE international journal of passenger cars-electronic and electrical systems*, 1(2008-01-0589):211–219, 2008.

[6] Reza Hoseinnezhad and Alireza Bab-Hadiashar. Efficient antilock braking by direct maximization of tire–road frictions. *IEEE Transactions on Industrial Electronics*, 58(8):3593–3600, 2011.

[7] Lei Yuan, Hong Chen, Bingtao Ren, and Haiyan Zhao. Model predictive slip control for electric vehicle with four in-wheel motors. In *2015 34th Chinese Control Conference (CCC)*, pages 7895–7900. IEEE, 2015.

[8] Wei-Yen Wang, I-Hsum Li, Ming-Chang Chen, Shun-Feng Su, and Shi-Boun Hsu. Dynamic slip-ratio estimation and control of antilock braking systems using an observer-based direct adaptive fuzzy–neural controller. *IEEE Transactions on Industrial Electronics*, 56(5):1746–1756, 2009.

[9] Ricardo De Castro, Rui E Araújo, Mara Tanelli, Sergio M Savaresi, and Diamantino Freitas. Torque blending and wheel slip control in evs with in-wheel motors. *Vehicle System Dynamics*, 50(sup1):71–94, 2012.

[10] Marcel Stefan Geamanu, Arben Cela, Guénaël LeSolliec, Hugues Mounier, and Silviu-Iulian Niculescu. Maximum friction estimation and longitudinal control for a full in-wheel electric motor vehicle. In *2012 12th International Conference on Control, Automation and Systems*, pages 856–861. IEEE, 2012.

[11] Jia-Sheng Hu, Ying-Ruei Huang, and Feng-Rung Hu. Development of traction control for front–wheel drive in–wheel motor electric vehicles. *International Journal of Electric and Hybrid Vehicles*, 4(4):344–358, 2012.

[12] Yu Tian and Nilanjan Sarkar. Control of a mobile robot subject to wheel slip. *Journal of Intelligent & Robotic Systems*, 74(3-4):915–929, 2014.

[13] Hamza Khan, Jamshed Iqbal, Khelifa Baizid, and Teresa Zielinska. Longitudinal and lateral slip control of autonomous wheeled mobile robot for trajectory tracking. *Frontiers of Information Technology & Electronic Engineering*, 16(2):166–172, 2015.

[14] Mou Chen. Disturbance attenuation tracking control for wheeled mobile robots with skidding and slipping. *IEEE Transactions on Industrial Electronics*, 64(4):3359–3368, 2017.

[15] Juliano G Iossaqui and Juan F Camino. Wheeled robot slip compensation for trajectory tracking control problem with time-varying reference input. In *9th International Workshop on Robot Motion and Control*, pages 167–173. IEEE, 2013.

[16] Nathan Wallace, He Kong, Andrew Hill, and Salah Sukkarieh. Receding horizon estimation and control with structured noise blocking for mobile robot slip compensation. *CoRR*, abs/1810.04366, 2018.

[17] Haibo Gao, Xingguo Song, Liang Ding, Kerui Xia, Nan Li, and Zongquan Deng. Adaptive motion control of wheeled mobile robot with unknown slippage. *International Journal of Control*, 87(8):1513–1522, 2014.

[18] Ramon Gonzalez, Mirko Fiacchini, Teodoro Alamo, Jose Luis Guzman, and Francisco Rodriguez. Adaptive control for a mobile robot under slip conditions using an lmi-based approach. *European Journal of Control*, 16(2):144–155, 2010.

[19] Ramon Gonzalez, Francisco Rodriguez, Jose Luis Guzman, Cedric Pradalier, and Roland Siegwart. Control of off-road mobile robots using visual odometry and slip compensation. *Advanced Robotics*, 27(11):893–906, 2013.

[20] Tor A Johansen and Thor I Fossen. Control allocation—a survey. *Automatica*, 49(5):1087–1103, 2013.

[21] Marc Bodson and Susan Frost. Control allocation with load balancing. In *AIAA guidance, navigation, and control conference*, page 6270, 2009.

[22] Kristoffer Tagesson, Peter Sundstrom, Leo Laine, and Nicolas Dela. Real-time performance of control allocation for actuator coordination in heavy vehicles. In *2009 IEEE Intelligent Vehicles Symposium*, pages 685–690. IEEE, 2009.

[23] Russell P Osborn and Taehyun Shim. Independent control of all-wheel-drive torque distribution. *Vehicle system dynamics*, 44(7):529–546, 2006.

[24] Juyong Kang, Hyundong Heo, et al. Control allocation based optimal torque vectoring for 4wd electric vehicle. Technical report, SAE Technical Paper, 2012.

[25] Arash M Dizqah, Basilio Lenzo, Aldo Sorniotti, Patrick Gruber, Saber Fallah, and Jasper De Smet. A fast and parametric torque distribution strategy for four-wheel-drive energy-efficient electric vehicles. *IEEE Transactions on Industrial Electronics*, 63(7):4367–4376, 2016.

[26] Kenta Maeda, Hiroshi Fujimoto, and Yoichi Hori. Four-wheel driving-force distribution method for instantaneous or split slippery roads for electric vehicle. *automatika*, 54(1):103–113, 2013.

[27] Ryosuke Eto, Kazuomi Sakata, and Junya Yamakawa. Driving force distribution based on tyre energy for independent wheel-drive vehicle on rough ground. *Journal of Terramechanics*, 76:29–38, 2018.

[28] Cesar A. Lopez Martinez. TRL5 prototype of low-level motion controller for joint level and Cartesian level control of smart wheel. ROPOD project, jun 2018.

[29] ROPOD project: Preliminary Results. `http://www.ropod.org/preliminary_results.html`. Accessed: 06-06-2018.

[30] J.P.M. Verhagen. Odometry and localization estimation of a mobile robot with multiple swivel wheels. ROPOD project, jan 2019.

Technische Universiteit
Eindhoven
University of Technology

Declaration concerning the TU/e Code of Scientific Conduct for the Master's thesis

I have read the TU/e Code of Scientific Conduct[i].

I hereby declare that my **Master's thesis** has been carried out in accordance with the rules of the TU/e Code of Scientific Conduct

Date

07-05-2019

Name

F.C.A. Huijser

ID-number

0769026

Signature

Submit the signed declaration to the student administration of your department.

[i] See: http://www.tue.nl/en/university/about-the-university/integrity/scientific-integrity/
The Netherlands Code of Conduct for Academic Practice of the VSNU can be found here also.
More information about scientific integrity is published on the websites of TU/e and VSNU