

A Novel Approach For Steeringwheel Synchronization With Velocity/Acceleration Limits And Mechanical Constraints

Ulrich Schwesinger, Cedric Pradalier and Roland Siegwart

Abstract—Pseudo-omnidirectional robots with independently steerable wheels require a method to synchronize the steering motion of the wheels in order to keep a unique instantaneous center of rotation (ICR). For standard wheels, the instantaneous center of rotation is defined as the intersection point of all wheel axes. We present a novel approach to deal with the problem of continuously shifting the center of rotation of a pseudo-omnidirectional rover from an initial to a demanded position in the Cartesian plane. The main contribution is the consideration of substantial velocity and acceleration limits on the steering units, as well as mechanical constraints and noise affected sensor measurements. We solve this problem by deriving a relationship between the steering accelerations of the single wheels and the acceleration of the center of rotation. We furthermore provide a contribution to the tracking of the ICR in the presence of significant sensor noise. Our results are evaluated by tests on the rover breadboard developed during the activities for the ExoMars mission.

I. INTRODUCTION

Robots with independently steerable wheels offer superior maneuverability, most valuable in rough terrains. However, without a sophisticated synchronization method of the wheels, misalignments can occur which may result in lower efficiency locomotion and power usage. Ideally, the rover's wheels ought to form a unique instantaneous center of rotation (ICR) at all times. If all intersection points coincide, the rover is moving on a well-defined trajectory. Misalignments arise due to the fact that a system with more than two independently steerable wheels is over-actuated [1]. This is intuitively obvious, since the ICR is already uniquely defined by two wheels. When the wheels are misaligned, friction, slip and internal forces will arise, leading to higher energy consumption. In space exploration scenarios such as ExoMars, available energy is limited and must be conserved. Since Mars is approximately one and half times further away from the Sun than the Earth, the energy collectable by solar panels is significantly reduced. In order to reduce the mass of the actuators, while maintaining sufficient torque capabilities, one might raise the gear reduction ratios of the steering units. However, this measure is traded for lower velocities and accelerations of the actuators. Low acceleration limits put a substantial difficulty on the problem of maintaining a unique ICR during a steering maneuver. The main problem we tackle in this paper is how to continuously move the ICR of the rover, while accounting for acceleration and velocity limits of the steering actuators at all times. Our approach is able to consider different kinds of velocity and acceleration profiles of the actuators and incorporates them in the calculations of the actuator commands. It can be regarded as a low-level

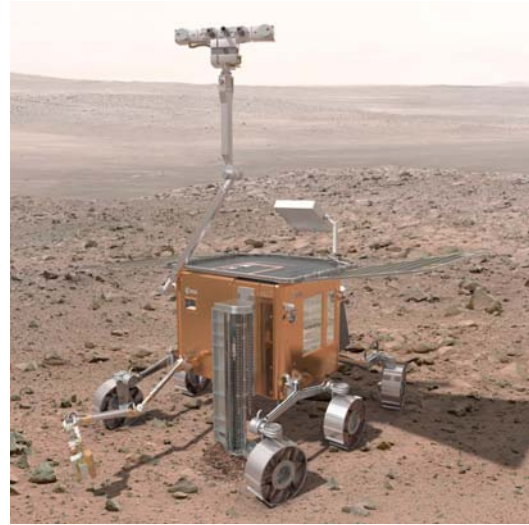


Fig. 1. ExoMars rover - phase B1 concept, source: ESA/Cluster

controller module for steering synchronization on top of the single wheels' motor controllers.

Mechanical limits of the steering units create zones in the ICR-space, that cannot be left unless at least one wheel rotates by 180 degrees [2]. Nevertheless, one zone can be connected over infinity in the cartesian plane. Thus, a discontinuity arises when moving the ICR from the right half-plane to the left or vice versa, which can cause problems within algorithms that control the rover's movement in the ICR-space. However, this movement of the ICR is totally permissible. It corresponds to a switching of the half-planes by going over a parallel wheel configuration as displayed in figure 2. In this specific figure, the steering angles are limited to the range of $[-\pi/2, \pi/2]$ as it was the case for the ExoMars platform. This creates the two different zones as shown. It is impossible to move the ICR from outside the rover frame to the inside without having to re-orient at least one wheel by 180 degrees. To simplify things, we therefore consider two different kinds of driving modes: an Ackermann mode, which is characterized by having the ICR outside the rover frame, and a point turn mode, where the ICR lies within.

The literature referring to the problem we address in this paper is rare. To the best of our knowledge, no previous work has addressed the problem of ICR control with acceleration limits. Moore et al. [3], designed a feedback linearization controller to track a given trajectory of an

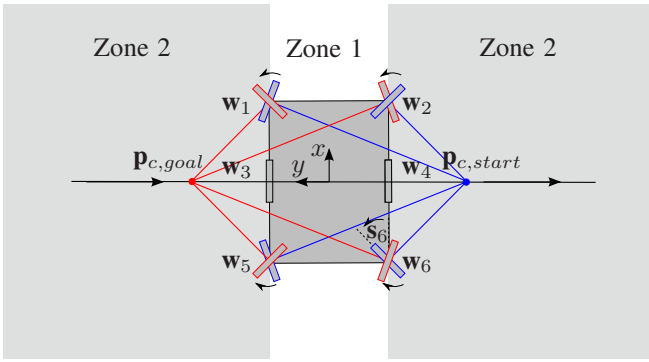


Fig. 2. Example of two ICR-zones introduced by mechanical constraints. The zones are connected over infinity. Entering e.g. “Zone 1” from “Zone 2” is not possible without re-orienting one wheel by 180° .

omni-directional rover by deriving a kinematic model of it, but did not account for any actuator limits or mechanical constraints. Connette et al. [4], utilized a model predictive controller to introduce constraints in the velocity space of the ICR. Acceleration constraints were not in the author’s focus.

In section II we will describe our strategy of moving the ICR without violating the velocity and acceleration limits of the actuators. We will also explain our approach to avoid mechanical limits and present a control strategy to ensure a smooth arrival at the target ICR. The tracking of the current ICR’s position and velocity in the presence of sensor noise is explained in section III, followed by the results of our tests on the ExoMars¹ breadboard in IV.

II. STEERING METHOD

Our goal is to synchronize the rover’s steering motion, that is, to ensure that the ICR is well-defined during all steering maneuvers. We do not consider a path tracking of the rover in a global frame and regard steering and driving motion separately. Our algorithm can be used in conjunction with a higher-level path-tracker that produces steering commands to follow a global path. We consider the input to our algorithm to be a desired ICR defined in the rover’s body frame. To guarantee the compliance of the motion of the ICR with the limits of the actuators, our approach is to explicitly derive the relation between the velocity/acceleration of the ICR and those of the actuators. Actuator limits can then be expressed as limits on the dynamics of the ICR.

A. Linking Steering Acceleration and ICR Acceleration

We define the position of the ICR with respect to the rover’s body frame in the Cartesian plane at time t with $\mathbf{p}_c(t) = [x(t), y(t)]^T$. The steering angle of wheel i corresponding to the ICR $\mathbf{p}_c(t)$ can then be expressed via

$$\mathbf{s}_i(t) = -\arctan\left(\frac{\mathbf{p}_{c,1}(t) - \mathbf{w}_{x,i}}{\mathbf{p}_{c,2}(t) - \mathbf{w}_{y,i}}\right), \quad (1)$$

¹More information about the ExoMars mission can be found on the ESA homepage at <http://exploration.esa.int/>

where \mathbf{w}_x and \mathbf{w}_y define the Cartesian coordinates of the wheels, with $i \in \{1, 2, \dots, N\}$ and $N \geq 3$ the number of steerable wheels. Steering angles are defined w.r.t. to the rover’s body frame. In our case the body frame is a right-handed coordinate system with its x-axis pointing in rover forward direction and its y-axis pointing to its left (see also Figure 3). Note that the inverse function, mapping a set of $N \geq 3$ steering angles to a unique ICR is not defined, since the $\binom{N}{2}$ intersection points of the wheel axes do not necessarily coincide. One can easily deactivate one or more wheels’ steering units during execution of the algorithm (it might be necessary due to a detected blockage of the wheel), just by disregarding these wheels in all calculations.

The representation of the ICR in Cartesian coordinates bears singularities in the case of parallel wheel configurations [2]. Then, the coordinates of the ICR lie on a circle with infinite radius. To avoid this singularity, the ICR is projected onto a unisphere with its support point in the center of the rover’s coordinate frame. By applying this transformation to spherical coordinates, the ICR can be represented by its azimuth θ and its elevation ϕ as depicted in figure 3. Note that these spherical coordinates are not the conventional ones, but a representation we chose. Now the ICR, when having a parallel wheel configuration, lies on the equator of the unisphere ($\phi = \frac{\pi}{2}$) with $\theta = \mathbf{s}_i + \frac{\pi}{2}$. \mathbf{s}_i is the common steering angle for all wheels.

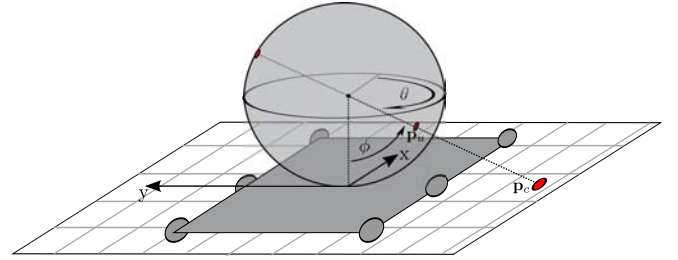


Fig. 3. Projection of the ICR onto a unisphere

When having ICR zones connected over infinity in the Cartesian plane, switching the half-plane corresponds to moving the ICR across the equator of the unisphere. The discontinuity in the Cartesian plane vanishes in the ϕ/θ -plane, and a continuous planning of the ICR’s motion is possible.

The equation of the steering angles depending on the position of the new ICR $\mathbf{p}_u(t) = [\phi(t), \theta(t)]^T$ located on the unisphere then transforms into

$$\mathbf{s}_i(t) = -\arctan\left(\frac{\cos(\mathbf{p}_{u,2}(t)) \tan(\mathbf{p}_{u,1}(t)) - \mathbf{w}_{x,i}}{\sin(\mathbf{p}_{u,2}(t)) \tan(\mathbf{p}_{u,1}(t)) - \mathbf{w}_{y,i}}\right). \quad (2)$$

For point turn mode, it is not necessary to perform this projection, since the ICR will always be within the rover frame and no singularities can arise in Cartesian coordinates. Additionally, the projection of the ICR onto the unisphere is not defined at $[x, y] = [0, 0]$, due to $\theta = \arctan(\frac{y}{x})$.

By building the first and second derivative of the steer-

ing angles with respect to time given in equation (2) for Ackermann mode, respectively equation (1) for point turn mode, one can derive the link between the steering speeds/accelerations of the steering units and the velocity/acceleration of the ICR.

The relationships between steering speeds ω , steering accelerations $\dot{\omega}$ and ICR velocity \mathbf{v} and acceleration \mathbf{a} are given by

$$\omega_i = \mathbf{J}_i(\mathbf{p}) \cdot \mathbf{v} = \begin{bmatrix} \frac{\partial s_i}{\partial p_1} & \frac{\partial s_i}{\partial p_2} \end{bmatrix} \cdot \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} \leq \Omega_{max} \quad \forall i, \quad (3)$$

$$\begin{aligned} \dot{\omega}_i &= \frac{\partial}{\partial t} (\mathbf{J}_i(\mathbf{p})) \cdot \mathbf{v} + \mathbf{J}_i(\mathbf{p}) \cdot \frac{\partial}{\partial t} (\mathbf{v}) \\ &= \mathbf{G}_i(\mathbf{p}, \mathbf{v}) \cdot \mathbf{v} + \mathbf{J}_i(\mathbf{p}) \cdot \mathbf{a} \leq A_{max} \quad \forall i, \end{aligned} \quad (4)$$

with $\mathbf{J}_i(\mathbf{p})$ being the i -th row of the Jacobian ($N \times 2$ matrix) of equation (1) or (2) at ICR position \mathbf{p} . $\mathbf{G}(\mathbf{p}, \mathbf{v})$ contains the derivatives of the entries of the Jacobian with respect to time and is dependent on the ICR position and velocity. The actuator's maximum speed and acceleration are given by Ω_{max} and A_{max} . \mathbf{J} and \mathbf{G} were computed with the aid of MAPLE, but are unfortunately too complex to be displayed here². They have to be recalculated in every time-step based on an estimation of the ICR's current position \mathbf{p} and velocity \mathbf{v} . We dropped the index of the ICR, describing the representation (either Cartesian or spherical) of the ICR so far, to indicate that these equations hold regardless of the transformation applied to the ICR.

Having two free parameters \mathbf{a}_1 and \mathbf{a}_2 , the direction of \mathbf{a} is freely selectable at this point. By specifying this direction of acceleration $\beta = \arctan\left(\frac{a_2}{a_1}\right)$ of the ICR, equation 4 can be evaluated to get the maximum permitted acceleration of the ICR \mathbf{a}_{max} , which respects the acceleration limit A_{max} of the steering units. The maximum permitted acceleration of the ICR follows from (4):

$$\mathbf{a}_{max,1} = \arg\max_i \left\{ \frac{A_{max} - \mathbf{G}_i(\mathbf{p}, \mathbf{v}) \cdot \mathbf{v}}{\mathbf{J}_{i,1} + \mathbf{J}_{i,2} \tan(\beta)} \right\}, \quad (5)$$

$$\mathbf{a}_{max,2} = \tan(\beta) \cdot \mathbf{a}_{max,1}. \quad (6)$$

We assume a constant A_{max} , implying velocity ramps of the actuators. However, different velocity profiles are possible by varying A_{max} depending on the actuators' current speeds. In order to obtain a feasible velocity and position of the ICR at the next time-step, \mathbf{a} is integrated with a bilinear integration over one sample period. To assure that the calculated velocity of the ICR complies with the maximum speed of the steering units, equation (3) is used to evaluate $\frac{\partial s}{\partial t}$. If the calculated velocity of the ICR should result in a violation of one or more wheels' maximum steering speeds, one can easily scale down the ICR's velocity due to the linear relationship to the steering speeds.

²The full equations can be found in the technical report at http://www.asl.ethz.ch/people/sculrich/exomars_technical_report.

B. Potential Field Approach to Determine Direction of ICR-Acceleration

We apply an artificial force on the ICR to determine its moving direction. This artificial force can be designed to avoid forbidden configurations or frontiers in the ICR-space [2], [4], [5].

It is important to prevent the ICR from crossing frontiers in the ICR space introduced by mechanical limits. Due to sensor noise and various imperfections of the system, it cannot be guaranteed by the algorithm itself, that these frontiers will not be traversed slightly. This is especially likely when operating close to these frontiers. Additionally, forbidden regions in the ICR-space exist at the origins of the wheel axes. Here, the steering angle of the specific wheel is not defined. In order to avoid these forbidden configurations, an artificial potential field, consisting of an attractive force originating from the target ICR and a repulsive force, was introduced [4]. We utilized this potential field approach only for point turn mode in Cartesian coordinates, since ExoMars-specific maximum curvature conditions prevented an operation close to ICR frontiers in Ackermann mode. Applying the potential field approach in the ϕ/θ -plane is in general possible. One might design the potential field - since easier to handle - in the x/y -plane, and transform it into the ϕ/θ -plane afterwards. In our implementation of point-turn mode, the repulsive force was designed to push the ICR towards the lateral center of the rover and was chosen to be linear with the lateral offset of the ICR. The repulsive force was reduced linearly with the distance to the target ICR, to decrease the influence when approaching the target. This potential field design avoids local minima of the potential function and turned out to be sufficient for our needs despite its simplicity. The attractive force of the

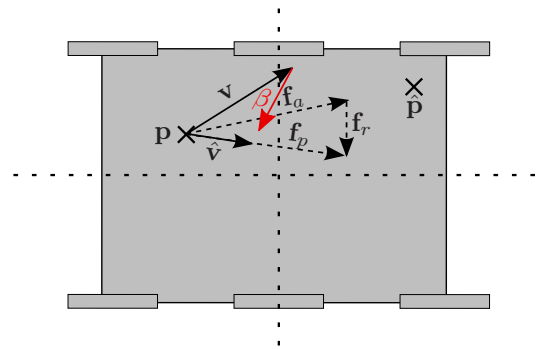


Fig. 4. Virtual forces \mathbf{f}_a and \mathbf{f}_r applied to the ICR to determine the desired velocity $\hat{\mathbf{v}}$ and the desired direction of acceleration β .

potential field points from the current position to the goal position. With the repulsive force of the potential field, we modify this direction to avoid frontiers in the ICR-space (see Figure 4). In our case we applied a purely lateral repulsive

force

$$\begin{aligned} \mathbf{f}_p &= \underbrace{\mathbf{f}_a}_{\text{attractive force}} + \underbrace{\mathbf{f}_r}_{\text{repulsive force}} \\ &= \hat{\mathbf{p}} - \mathbf{p} + \begin{bmatrix} 0 \\ -\epsilon \cdot \mathbf{p}_2 \end{bmatrix}, \end{aligned} \quad (7)$$

with $\hat{\mathbf{p}}$ being the target ICR and ϵ a factor to modify the strength of the repulsive force.

An exemplary potential field is shown in figure 5. It is apparent, that this potential field design is local minima free, however differing mechanical constraints might require another design [6], [7].

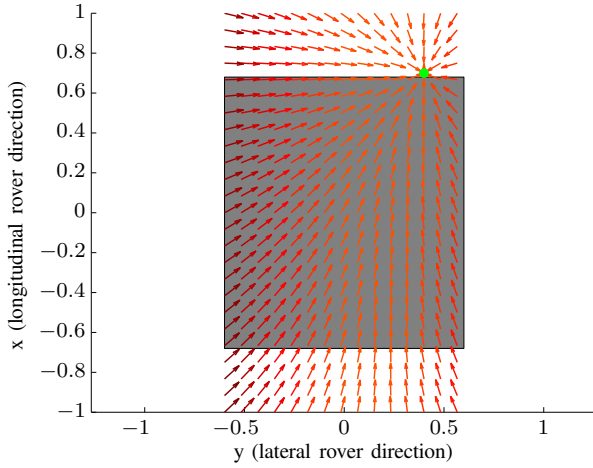


Fig. 5. Potential field designed for point turn mode. Grey box: rover chassis, green dot: target ICR, arrows: color-coded potential field forces (darker colors correspond to larger forces).

C. Control of the ICR in ICR-Space

When approaching the desired ICR $\hat{\mathbf{p}}$, it is obvious that a strategy is required to slow down the steering units. We chose to implement a control strategy in which the movement of the ICR is controlled by a nested P-Controller. This controller assures a smooth approach to the target ICR. Since the maximum permitted acceleration of the ICR is determined by the steering synchronization described above, a saturation function is introduced in the process. Therefore, the control is only active in close proximity to the target ICR, depending on the gains K_v and K_a . Wind-up effects are no issue, since no integral part in the controller is present. The closed loop control system is displayed in figure 6.

We applied a rectangular forward integration to the acceleration since the acceleration was considered to be constant between two sample times. Thus, for velocity we used a bilinear integration. The transfer function (neglecting the saturation function) can be determined to be

$$G(z) = \frac{K_v K_a T^2 (z+1)}{2z^2 + (K_v K_a T^2 - 4 + 2K_a T)z + K_a (K_v T^2 - 2T) + 2}. \quad (8)$$

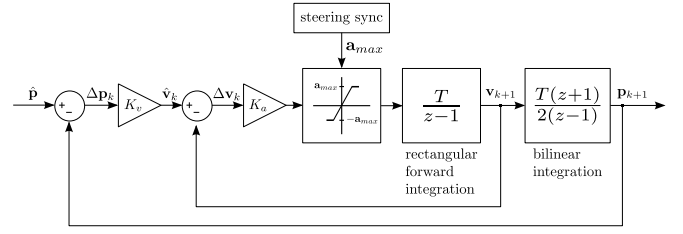


Fig. 6. Nested P-Controller to control the ICR's position and velocity

The controller is stable if the poles of the transfer function lie within the unit circle. An overshoot can be avoided by eliminating the imaginary part of the poles. The obtained value \mathbf{p}_{k+1} for the new position of the ICR in the next time-step can be converted into the corresponding steering angles with equation (1), respectively (2).

D. Calculation of Feasible Actuator Commands

Our algorithm produces desired steering angles s for every time-step. Most commercial motor controllers offer low-level position and velocity control modes. In principle, our algorithm has to run in velocity control, since sending the calculated steering angles in each time-step would cause the internal motor controllers to break down to come to standstill at the desired position. In this work, velocity ramp profiles, respectively rectangular acceleration profiles were assumed. Therefore, the previously calculated steering angle commands have to be transformed into steering speed commands ω_c , considering the velocity ramp profile shown in figure 7.

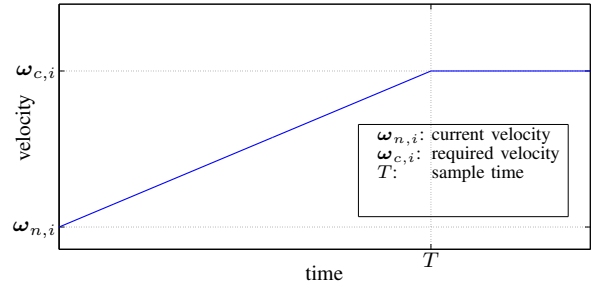


Fig. 7. Velocity ramp profile

One can calculate the steering speed required to reach the demanded position for wheel i via

$$\omega_{c,i} = \omega_{n,i} + A_{max} \cdot \left(T - \sqrt{T^2 + \frac{2(\omega_{n,i}T - \Delta s_i)}{A_{max}}} \right), \quad (9)$$

with T being the sample time, $\omega_{c,i}$ and $\omega_{n,i}$ the commanded and actual steering speed of wheel i , and Δs being the steering angle difference to the requested position in the next time-step.

III. TRACKING OF THE ICR

The steering synchronization method described above requires and strongly depends on a good estimate of the current ICR's position and velocity. As the sensor values regarding steering position and velocity are always noisy up to a certain amount, or the actual steering positions might deviate from the desired ones due to a partial blockage of the wheels, an extended Kalman filter (EKF) was implemented to estimate the ICR's position and velocity.

A. Setup of the Extended Kalman Filter

The state vector for the EKF contains the ICR's position and velocity, either in the ϕ/θ -plane (for Ackermann mode) or the x/y -plane (for point-turn mode):

$$\mathbf{x} = \left[\mathbf{p}, \frac{\partial \mathbf{p}}{\partial t} \right]^T. \quad (10)$$

The measurement vector \mathbf{z} consists of all steering angles and steering speeds, leading to $2N$ entries, where N is the number of steerable wheels:

$$\mathbf{z} = \left[\mathbf{s}, \frac{\partial \mathbf{s}}{\partial t} \right]^T. \quad (11)$$

The measurement matrix \mathbf{H} is built by the partial derivatives of the steering angles and steering speeds with respect to the ICR position and velocity

$$\mathbf{H} = \begin{bmatrix} \frac{\partial \mathbf{s}}{\partial \mathbf{x}} \\ \frac{\partial \frac{\partial \mathbf{s}}{\partial t}}{\partial \mathbf{x}} \end{bmatrix} = \begin{bmatrix} \mathbf{J} & \mathbf{0} \\ \mathbf{G} \cdot \mathbf{v} & \mathbf{J} \end{bmatrix}, \quad (12)$$

and forms a $[2N \times 4]$ matrix. The prediction of the state is taken from the controller output described above. For the prediction of the covariance matrix, the system matrix \mathbf{F} is required, which can be easily determined to be

$$\mathbf{F} = \begin{bmatrix} 1 & 0 & T & 0 \\ 0 & 1 & 0 & T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (13)$$

with sample time T . The measurement noise refers directly to the position and velocity noise of the actuators and a white noise assumption seems justified. The system noise most likely depends on various influences, like time-delays in the system and also on the current position and velocity of the ICR. Assuming a fixed white system noise yielded good results but might describe the true noise imperfectly.

B. Initialization of the Extended Kalman Filter

The initialization of the filter requires a first estimate of the ICR position and velocity. If this estimation is bad, the filter might diverge due to the linearizations made. We calculated a first estimate of the ICR's position by choosing the intersection point of two wheels, that leads to the minimum steering angle deviation variance for all wheels. This position is then refined with a standard Levenberg-Marquardt approach. This turned out to be sufficient to assure a fast convergence and to avoid divergence of the filter. However, a more sophisticated

approach leading to better estimations for near-parallel wheel configurations can be found in [8].

After having an initial guess for the ICR's position, the ICR's velocity is then obtained by a linear Least-Squares estimator, solving equation (3) for \mathbf{v} :

$$\mathbf{v} = (\mathbf{J}^T \mathbf{J})^{-1} \mathbf{J}^T \boldsymbol{\omega}. \quad (14)$$

IV. RESULTS

To evaluate the effectiveness and the benefit of synchronizing the steering wheels, tests on the rover breadboard developed for the ExoMars project were performed. The six-wheeled rover breadboard was driven in a sand testbed, designed to simulate the conditions on the Martian surface. Since none of the approaches described in [2] or [4] incorporates acceleration limits, and therefore a reasonable behavior cannot be guaranteed, we compare our **synchronized approach (SM)** to a **naive method (NM)**. The NM sets the wheels' steering speeds according to their remaining angular deviation to the target position with

$$\omega_{c,i} = \Omega_{max} \cdot \frac{\Delta \mathbf{s}_i}{\underset{i}{\operatorname{argmax}} \{ \Delta \mathbf{s}_i \}}. \quad (15)$$

The wheel with the largest deviation is commanded the maximum permitted steering speed Ω_{max} . This naive method leads to a concurrent arrival of the steering units at their target position in the absence of acceleration constraints.

Two predefined steering sequences were tested, one sequence each for Ackermann and point turn mode. Each steering sequence was repeated three times for the synchronized and the naive steering approach. The sequences were defined by time-stamped steering/driving events, which are given in Table I. The driving speed for Ackermann mode corresponds to the translational speed of the center of the rover, in point turn mode it defines the rotational speed around the current ICR. The steering sequences were designed to perform maximum steering motion with the space available in the testbed. They also comply with the ExoMars-specific maximum curvature constraints. Because of these constraints, no repulsive potential field was applied in Ackermann mode, since the constraints prevented an operation near the mechanical limits anyways. In both cases the wheel driving speeds were synchronized with respect to the estimated ICR.

As a quantitative measure of performance, the lateral forces provided by three force sensors mounted on the drive axes on the right side of the rover were recorded. Current readings were available for all six drive axes. In addition, the root mean square error (RMSE) of the steering angles to the perfect synchronization was calculated. The perfect synchronization was defined by the ICR estimated by the Levenberg-Marquardt Algorithm. The sample rate of the algorithms was set to $f_s = 5Hz$ to demonstrate the good performance despite lower update rates. In the following, the abbreviations FR, FL, CL, CR, RL and RR in the figures' legends will indicate the rover's six wheel positions

event in s	Ackermann			point turn		
	ICR _x in m	ICR _y in m	speed in m/s	ICR _x in m	ICR _y in m	speed in rad/s
1	0.00	∞	0.02	0.00	0.00	0.00
10	0.00	+0.70	0.02	+1.00	0.00	+0.01
20	0.00	-0.70	0.02			
30				-1.00	0.00	-0.01
40	0.00	+0.70	0.02			
60	0.00	-0.70	0.02	+1.00	+0.30	+0.01
80	+0.92	+1.09	0.02			
90				-1.00	-0.30	-0.01
100	-0.92	+1.09	0.02			
120	+0.92	-1.09	0.02			
130				0.00	0.00	+0.01
140	-0.92	-1.09	0.02			
150				0.00	0.00	0.00
160	0.00	∞	0.02			
170	0.00	∞	0.00			

TABLE I
SEQUENCE OF INPUT COMMANDS FOR TEST RUNS

(F)ront/(C)enter/(R)ear - (L)eft/(R)ight. Table II provides the values for maximum steering speed, acceleration and the rover dimensions.

steering actuator limits		rover dimensions	
A_{max}	Ω_{max}	rover width	rover length
$0.0302 \frac{\text{rad}}{\text{s}^2}$	$0.16 \frac{\text{rad}}{\text{s}}$	1.2 m	1.36 m

TABLE II
ROVER SPECIFIC SETTINGS

In figure 8 and 9, the lateral forces on the wheels and the RSME of the steering alignment are compared for a point-turn run. Each time, the upper plot displays the SM. One can observe a decrease of approximately 26% of the mean forces on the wheels, as well as a significant reduction in the RMSE for the SM.

The progress of the steering angles is depicted in figure 10. This time, the Ackermann maneuver was chosen, since the synchronization of the wheels is clearly visible when the steering angles pass the parallel configuration, when switching the half-planes of the ICR. In general, the duration to reach the target ICR is prolonged for the SM, so we are trading the benefit of lower internal forces for a slower steering movement. This is due to the fact that for SM, the wheel with the maximum steering angle difference to the target position is not always allowed to steer fastest, which can be observed by the smoothed plateau in figure 10 for the SM.

Table III shows the relation of integrated absolute values of driving currents over the whole run with respect to the maximum value. Mean values of the three runs performed were taken. For Ackermann mode, no significant reduction in energy consumption could be observed. However, for point turn mode, the SM is able to reduce the energy consumption by 16.04%. This discrepancy between Ackermann mode and

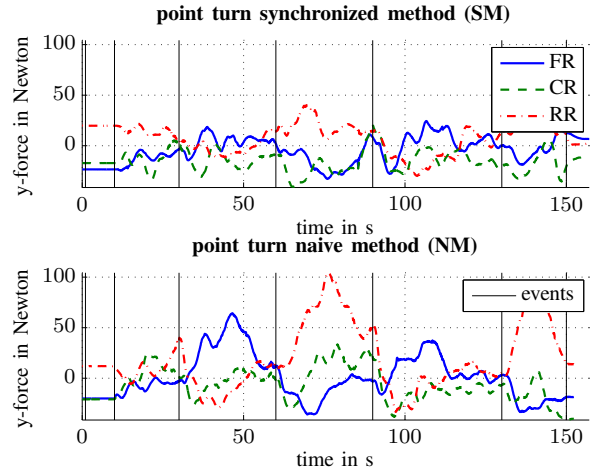


Fig. 8. Lateral forces recorded during point turn maneuver. The wheel synchronization results in a noticeable reduction of lateral forces on the wheels. Black event lines mark the occurrence of new ICR commands. The synchronization reduces the mean side-forces on the wheels by 26%.

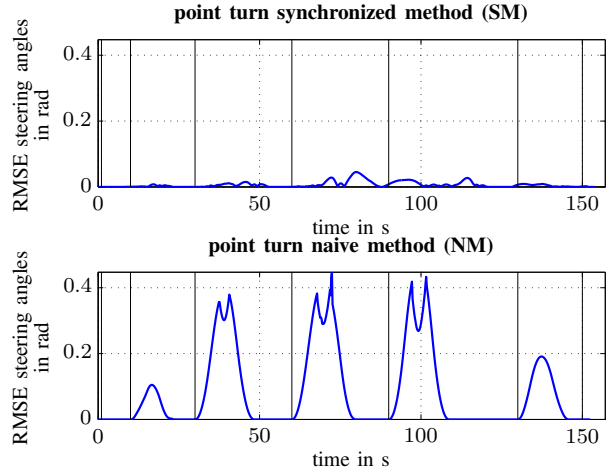


Fig. 9. Root mean square error of steering angles recorded during point turn maneuver.

point turn mode is due to the fact, that steering motions were wider for point turn mode and the misalignment of the steering units became larger during the motion.

	Ackermann	Point Turn
	$\int I \cdot dt$	
synchronized:	100 %	83.96 %
naive:	98.57 %	100 %

TABLE III
INTEGRATED DRIVE CURRENTS FOR ACKERMANN AND POINT TURN
MANEUVER

The effect of a blocked wheel could only be evaluated in simulation. Simulation results showed that the SM for the unblocked wheels is still working; the unblocked wheels keep a unique ICR. However, as an effect of the Kalman filtering, the target ICR is never reached completely. The

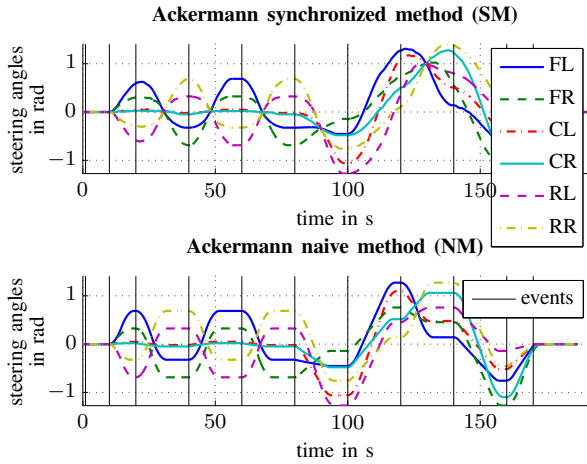


Fig. 10. Steering angles recorded during Ackermann maneuver. Synchronization is mainly visible at the intersection of the six steering angles in the upper plot, referring to parallel wheel configurations when moving the ICR from one side of the rover to the other.

measurements of the blocked wheel tend to pull the estimated ICR towards the blocked wheel, due to the averaging properties of the Kalman filter. However, this problem could easily be avoided by deactivating a wheel based on its deviation to the estimated ICR.

Finally, Figure 11 shows the path of the ICR for a simulated point turn maneuver. The influence of the repulsive force of the potential field results in a deformed path, avoiding ICRs close to lateral side of the rover.

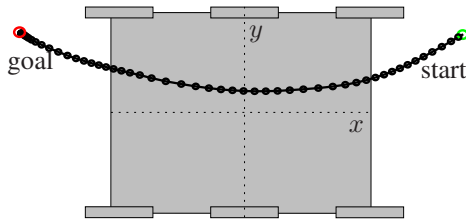


Fig. 11. ICR trajectory during simulated point turn maneuver. The potential field pushes the ICR towards the lateral center of the rover. Markers are placed at equal time increments.

V. CONCLUSION

We presented a novel approach for steering wheel synchronization in the presence of significant velocity and acceleration limits. We achieved this by projecting the constraints

of the steering units into the space of the instantaneous center of rotation. The ICR could then be controlled in a system-compliant manner. Our approach is able to deal with reasonable sensor noise through the filtering of steering angle and steering speed measurements with an EKF. In the case of large steering motions, the steering synchronization showed measurable benefits in the energy consumption of the drive units due to lower internal forces and slip. We believe that savings in energy would even be larger in applications where overactuated vehicles have to drive on harder surfaces than the sandy soil we had available for our evaluation. Undesirable side forces, that can be avoided with our approach, would become more severe on firmer ground. Furthermore, the algorithm showed good results in simulation with blocked wheels, which is an important property for rough terrain operation.

Further investigations could tackle the problem of calculating the ICR trajectory, that leads to the fastest possible arrival at the target ICR or to incorporate the steering synchronization into a global trajectory-tracking algorithm.

VI. ACKNOWLEDGEMENTS

This work was supported by ESA and RUAG Space AG in the context of the the ExoMars project, within which RUAG is responsible for the Locomotion Subsystem, Astrium Ltd is responsible for the Rover Vehicle and Thales Alenia Space Italy is the overall mission prime contractor to ESA.

REFERENCES

- [1] P. Muir and C. P. Neuman, "Kinematic modeling of wheeled mobile robots," tech. rep., Robotics Institute, 1986.
- [2] S. Chamberland, E. Beaudry, L. Clavien, F. Kabanza, F. Michaud, and M. Lauria, "Motion planning for an omnidirectional robot with steering constraints," in *Intelligent Robots and Systems (IROS)*, 2010.
- [3] K. Moore, M. Davidson, V. Bahl, S. Rich, and S. Jirgal, "Modeling and control of a six-wheeled autonomous robot," in *American Control Conference*, 2000.
- [4] C. Connette, A. Pott, M. Hagele, and A. Verl, "Addressing input saturation and kinematic constraints of overactuated undercarriages by predictive potential fields," in *Intelligent Robots and Systems (IROS)*, 2010.
- [5] C. P. Connette, C. Parltitz, M. Haegele, and A. Verl, "Singularity avoidance for over-actuated, pseudo-omnidirectional, wheeled mobile robots," 2009.
- [6] O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," *Robotics and Automation (ICRA)*, 1986.
- [7] C. Connolly, "Path planning using Laplace's equation," *Robotics and Automation (ICRA)*, 1990.
- [8] L. Clavien, M. Lauria, and F. Michaud, "Instantaneous centre of rotation estimation of an omnidirectional mobile robot," in *Robotics and Automation (ICRA)*, 2010.