



HAL
open science

Quasi-static motion simulation and slip prediction of articulated planetary rovers using a kinematic approach

Faïz Ben Amar, Christophe Grand

► **To cite this version:**

Faïz Ben Amar, Christophe Grand. Quasi-static motion simulation and slip prediction of articulated planetary rovers using a kinematic approach. *Journal of Mechanisms and Robotics*, 2013, 5 (2), pp.021002. 10.1115/1.4023873 . hal-03177962

HAL Id: hal-03177962

<https://hal.science/hal-03177962>

Submitted on 24 Mar 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Quasi-static motion simulation and slip prediction of articulated planetary rovers using a kinematic approach

Faiz Benamar, Christophe Grand

Institut des Systèmes Intelligents et de Robotiques
Université Pierre et Marie Curie Paris 6, CNRS UMR 7222
4, place jussieu, 75252 Paris Cedex 05
Email: [amar,grand]@isir.upmc.fr

Wheel slips are unavoidable when moving on a 3D rough surface. They are mainly due to geometrical features of contact surfaces. In this paper, we propose a model for predicting rover motion and contact slips by using a kineto-static model coupled with a linear contact model derived from semi-empirical tire/terramechanics approaches. The paper introduces also a coherent approach for motion simulation of uneven articulated rovers which is computationally efficient and can then be used for autonomous on-line path planning. Model results are compared to another numerical model based on a multi-body dynamic model including frictional contacts. The well-known rocker-bogie chassis, a highly articulated structure, is chosen to illustrate results and their comparison. Results demonstrate that for a slow motion, the proposed model approximates with a good accuracy the general behavior of the robot with a minimal time computation.

1 Introduction

Off-road mobile robotics is a new growing field which is made famous around the world by the recent NASA missions to Mars. Despite the success of these missions, many scientific challenges must be overcome before a real autonomous long-term navigation can be considered. One must notice that for these missions the robot executes autonomously a safe short-range path which is prepared and planned carefully on earth by human operators. Off-road mobile robotics still needs tools for modeling these systems which are characterized by a close interaction with a complex unstructured environment. Rover structures are multi-body articulated systems that interact with a ground surface through a set of frictional unilateral contacts. Rover modeling is highly useful for mechanical design, for designing control, for mission planning or path unmanned search. Two approaches for simulation can be distinguished : (i) dynamic-based approach and (ii) kineto-static based approach.

Dynamic approach The dynamic approach is more realistic than static one as they consider all forces exerted on the system, especially inertial forces of rover parts, visco-elastic forces in spring-damper and wheel-ground contact forces. These last ones are of high importance: the choice of a wheel-ground model deeply impacts the simulation results. Most contributions use Terramechanics models developed initially by Bekker [1], continued later by Wong [2] and improved recently by Iagnemma et al. [3]. Based on these models and using multi-body dynamics engine, some rover motion simulators are already developed, mainly those described in [4], [5] and [6]. Other studies use commercial physical dynamic simulator (e.g. Adams) for motion analysis [7] [8]. Unilateral wheel-ground contacts used in these CAD tools are generally considered as a force element and are based on : (1) a spring-damper model for computing the normal force and (2) a regularized Coulomb friction model for computing tangential force. Andrade et al. [9] use a different approach based on a contact discretization and on a granular ground model for computing matter flows when crossing non-cohesive sandy slopes. All of these models introduce many parameters for contact modeling, characterizing the non-linear elasto-plastic ground behaviour in both normal and tangential directions, as stiffness coefficient, stiffness exponent, cohesion, friction angle, shearing modulus, etc. The complexity of the dynamic approach and the multiplicity of contact parameters requires a significant computation time for the analysis of the rover performance. In contrast, for certain conditions the computation burden can be alleviated by using simplifying assumptions while maintaining the error at acceptable levels.

Kineto-static approach : From grasp to locomotion For spatial applications, the speed is quite low (less than 10 cm/s) validating the assumption of quasi-static motion neglecting thereby all inertial forces with respect to gravitational forces. The frequencies of ground surface variations are assumed

to be much lower than the lowest natural frequency of the system. On the one hand, static model is mainly used for wheel torques control [10] or for the computation of an optimal distribution of contact forces [11]. The problem of force distribution is also discussed in [12] and [13] for stability and mobility optimization in active wheeled-legged rovers. On the other hand, kinematic models are developed in [14] for pose computing using homogeneous transformation matrices and their time-derivative. Aucter et al. [15] adapts the concept of dexterous manipulation introduced initially by Li et al. [16], for kinematic motion simulation of a three wheeled rover. Their model introduces all kinematic contact parameters and their relationship to the surface parameters. The main drawback of this model is that it assumes that the contact point position on the surface is continuous with respect to time, although this is not a valid assumption when crossing a step and more generally when traveling on a 3D ground surface. Chakraborty et al [17] use a similar approach for the contact kinematic model and an equivalent hybrid-parallel mechanism for motion simulation of three-wheeled rover with variable length axle.

Solving coupled equilibrium-contact models Computation of contact forces permits the evaluation of physical validity of the rover pose i.e. its static stability and propulsion capacity. It is well known that most articulated multibody systems with unilateral contacts have a high degree of static indeterminacy, often called as hyperstaticism degree, that is the number of unknown force components is greater than the number of static equilibrium equations. In the field of structures computing, the static indeterminacy is solved by adding to the equilibrium equations the displacement compatibility equations and equations characterizing the rheological properties of the material (e.g. the Hooke's elasticity law). For our multi-body structure, its mechanical behaviour is mainly dependent on the behaviour of the wheel-ground contact and the representation of this interaction. However, this interaction is very complex and introduces many physical and mechanical phenomena such as tire elasticity, ground material elasto-visco-plasticity, ground material flow, grousers shearing, etc.

Articulated rovers have two distinct sources of static indeterminacy: internal and external. The internal hyperstaticism is due to the use of redundant actuation (all wheel drive). One must notice that these internal forces i.e. wheel torques are directly related to the longitudinal force components by the wheel radius. On the other hand, the external hyperstaticism is due to lateral components of contact forces and specially to the use of multi-wheel supports. Generally, the Moore-Penrose pseudo-inverse matrix is used to solve over or under-determined linear equation systems. Moreover, the obtained solution can have little physical meaning, especially when the constraints or variables are heterogeneous.

Trinkle [18] uses the Peshkin's minimum power princi-

ple [19] for quasi-static grasping systems to combine force and kinematic relationships into a nonlinear mathematical program called the forward object motion problem. This problem can be stated as follows: for given joint rates and contact parameters at a given time, determine the velocity of manipulated object with respect to the ground and the corresponding contact forces which ensure quasi-static equilibrium. However, the problem is in the form of a nonconvex, nonsmooth and nonlinear optimization program that can not be solved easily within a reasonable computing time.

Another approach developed in [20] introduces the assumption of "zero interaction". It assumes that all contact tangential forces work along the same direction (i.e. the equi-projectivity of the field of contact tangential forces are supposed to be coplanar). Obviously, this assumption minimizes internal forces and gives a solution comparable to the least-square one.

Proposed method This paper introduces a new method for solving the forward motion problem by using the kineto-static (differential kinematic and static) model of the articulated system and a linear contact model for wheel-ground characterization. The model will be applied to the motion simulation of articulated wheeled rovers. Kineto-static models characterizing velocity and force transmission between task, joint and contact frames will be established in section 2. Rover structures considered in this model can be uneven including fully multi-serial chains and branched trees with or without closed loops. The kinematic structures are first considered to be composed by serial chains, and next algebraic constraints are added to account for branches and closed loops. Section 3 discusses the friction model of the wheel-ground contact and proposes a simplified linear form of this model which includes two slip-velocity parameters. Section 4 proposes a resolution method of the forward motion problem based on: 1) a prediction-phase of normal contact forces based on the pseudo-inverse solution; and 2) a linear formulation of the forward motion problem for computing the robot velocity parameters, wheel slips and a more precise solution of the contact forces. The last section presents simulation results for different ground surface geometries, that are compared to other results provided by a dynamic simulation software.

2 Kineto-static model of articulated rovers

This section draws the kineto-static relationships of articulated rovers by introducing transformation matrices between 3 frames: joint, contact and task. The model will be developed first for a rover composed by a fully parallel structure. Next, constraint equations will be added to take into account the actual kinematics of the rover.

2.1 Velocity model of a multi-serial structure

Let's consider a robot structure with numerous serial kinematic chain going from a reference body called the platform and ending by a wheel which is in contact with the ground at point C . We assume for instance that the kinematic chains are independent and form a fully parallel structure. Let \mathcal{P} , \mathcal{W} , \mathcal{C} and \mathcal{G} be frames attached to the platform, the wheel, the contact and the ground respectively.

First, we will represent the contact between the wheel and the ground as a point contact between two rigid bodies. The contact coordinate frame \mathcal{C} is defined such that:

- the z -axis of this frame points in the direction of the outward surface normal at the contact point,
- the x -axis is the longitudinal tangent vector and can be computed by $\mathbf{x} = \frac{\mathbf{e} \times \mathbf{z}}{\|\mathbf{e} \times \mathbf{z}\|}$, where \mathbf{e} is a unit wheel axis vector.
- and y -axis is the lateral tangent vector defined by $\mathbf{y} = \mathbf{z} \times \mathbf{x}$.

Screw theory [21] allows us to write the twist relationship

$$\mathcal{S}_{gw} = \mathcal{S}_{gp} + \mathcal{S}_{pw} \quad (1)$$

where \mathcal{S}_{ab} is the twist of the frame b with respect to the frame a .

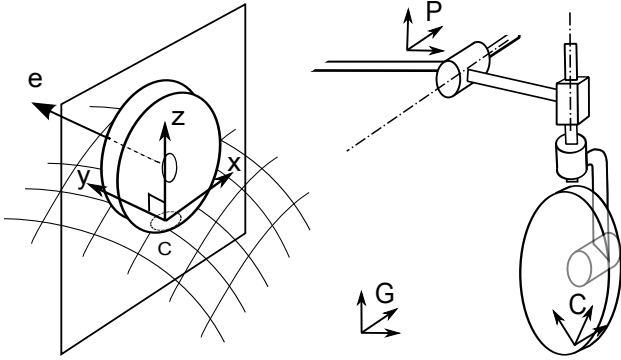


Fig. 1. Serial kinematic chain

This relationship is frame independent. In the following, we depict by $[\mathcal{S}_{ab}]_M^d$ the twist \mathcal{S}_{ab} computed at the point M and expressed in the coordinate system d (Appendix A).

2.1.1 Platform twist with respect to the ground

Platform configurations relative to ground can be parameterized by $\mathbf{x}_p = (x_p, y_p, z_p, o_x, o_y, o_z)^T$ which regroups the position of $O_p = (x_p, y_p, z_p)^T$ and the usual roll, pitch and yaw angles $(o_x, o_y, o_z)^T$ (Appendix B). $\dot{\mathbf{x}}_p$ is related to the twist $[\mathcal{S}_{gp}]_{O_p}^p$ by

$$\mathbf{v}_p = [\mathcal{S}_{gp}]_{O_p}^p = \mathbf{A}_{pg} \dot{\mathbf{x}}_p \quad (2)$$

where \mathbf{A}_{pg} is an invertible matrix, except for null pitch angle o_y , and its expression is given in Appendix B.

As will be showed later, it is convenient to express this twist entirely in the contact frame. Then, the adjoint matrix will be used

$$\begin{aligned} [\mathcal{S}_{gp}]_{O_c}^c &= \underbrace{\begin{pmatrix} \mathbf{R}_{cp} & \hat{\mathbf{u}}_{cp} \mathbf{R}_{cp} \\ \mathbf{0}_{3 \times 3} & \mathbf{R}_{cp} \end{pmatrix}}_{\text{Ad}_{\mathbf{T}_{cp}}} \underbrace{[\mathcal{S}_{gp}]_{O_p}^p}_{\mathbf{v}_p} \\ &= \text{Ad}_{\mathbf{T}_{cp}} \mathbf{v}_p \end{aligned} \quad (3)$$

where $\hat{\mathbf{u}}_{cp}$ is the skew-symmetric matrix associated to the vector cross-product, $\mathbf{u}_{cp} = \mathbf{r}_{O_c O_p}^c$ expressed in the contact coordinate system, and \mathbf{T}_{cp} is the homogeneous transformation matrix

$$\mathbf{T}_{cp} = \begin{pmatrix} \mathbf{R}_{cp} & \mathbf{u}_{cp} \\ \mathbf{0}_{1 \times 3} & 1 \end{pmatrix} \quad (4)$$

2.1.2 Wheel twist with respect to the platform

Let's consider a serial kinematic chain going from the platform to the wheel. This serial chain i has n_i joints which can be either revolute or prismatic and parameterized by θ_j or λ_k respectively. The wheel twist, w.r.t. the platform, can be computed in the same way as for a fixed base manipulator, except that this twist will be expressed at the contact point which is not a body attached point. Thus, this twist can not be given by time-derivative of geometric equations derived from homogeneous transformation matrices. Nevertheless, we can write

$$[\mathcal{S}_{pw}]_{O_c} = \begin{pmatrix} \dots & \mathbf{e}_j \times \mathbf{a}_j & \dots & \mathbf{e}_k & \dots \\ \dots & \mathbf{e}_j & \dots & \mathbf{0}_3 & \dots \end{pmatrix} \begin{pmatrix} \vdots \\ \dot{\theta}_j \\ \vdots \\ \dot{\lambda}_k \\ \vdots \end{pmatrix} \quad (5)$$

where $j, k = (1 \dots n_i)$ denote respectively the indexes of the revolute and the prismatic joints, $\mathbf{e}_j, \mathbf{e}_k$ are unit vectors of the joint axes, and \mathbf{a}_j is the vector going from a point located on the revolute joint axis to the contact point O_c . Note that this relation is system-coordinate independent. In general, these vectors are more easily expressed in the platform frame. As all scalar equations of the proposed model will be expressed in the contact frame, these vectors are projected in this frame by using the rotation matrix $\mathbf{R}_{cp} = \mathbf{R}_{pc}^T$. Finally, the wheel twist with respect to the platform is written by

$$[\mathcal{S}_{pw}]_{O_c}^c = \mathbf{J}_{pw} \dot{\boldsymbol{\theta}} \quad (6)$$

\mathbf{J}_{pw} is called Jacobian matrix. It must be noticed, that this designation is improper as this matrix has not got the

structure of a partial derivative matrix as for a simple usual manipulator where the Forward Geometrical Model is expressed in a fixed-body point attached to the end-effector. In our case, the wheel point O_c is an instantaneous point that can not be tracked by usual time-coordinate (non-holonomic system).

2.1.3 Differential kinematic model of a multi-serial kinematics

Let $[\$_{gw}]_{O_c}^c$ the wheel-ground twist expressed in the local contact frame. Substituting equations (2) and (5) into (1), we obtain

$$[\$_{gw}]_{O_c}^c = \text{Ad}_{\mathbf{T}_{cp}} \mathbf{v}_p + \mathbf{J}_{pw} \dot{\theta} \quad (7)$$

When multiplying this equation by

$$\mathbf{B}_{gw} = (\mathbf{1}_{3 \times 3} | \mathbf{0}_{3 \times 3}) \quad (8)$$

we obtain the linear component of this twist equation

$$\begin{aligned} \mathbf{s} &= \mathbf{B}_{gw} [\$_{gw}]_{O_c}^c \\ &= \mathbf{B}_{gw} \text{Ad}_{\mathbf{T}_{cp}} \mathbf{v}_p + \mathbf{B}_{gw} \mathbf{J}_{pw} \dot{\theta} \end{aligned} \quad (9)$$

with $\mathbf{s} = (s_x, s_y, s_z)^T$, the contact relative velocity. The three angular equations in this twist equation are not really interesting since they introduce three unknown angular parameters which are passive parameters and generally need to be eliminated in an input/output kinematic model.

This last equation can be written in a more compact form

$$[\mathbf{L}^i]^T \mathbf{v}_p + \mathbf{J}^i \dot{\theta}^i = \mathbf{s}^i \quad (10)$$

where i is the index of the kinematic chain (and it ranges from 1 to n_w , where n_w depicts the number of wheels), $\mathbf{L}^i = \text{Ad}_{\mathbf{T}_{cp}}^T \mathbf{B}_{gw}^T$ and $\mathbf{J}^i = \mathbf{B}_{gw} \mathbf{J}_{pw}^i$.

Bringing together the n_w equations, the differential kinematic model of the whole fully parallel articulated structure takes the following form:

$$\mathbf{L}^T \mathbf{v}_p + \mathbf{J} \dot{\theta} = \mathbf{s} \quad (11)$$

where

$$\mathbf{L}^T = \begin{pmatrix} [\mathbf{L}^1]^T \\ \vdots \\ [\mathbf{L}^{n_w}]^T \end{pmatrix} \quad (12)$$

$$\mathbf{J} = \begin{pmatrix} \mathbf{J}^1 & \dots & \mathbf{0} \\ \vdots & \ddots & \vdots \\ \mathbf{0} & \dots & \mathbf{J}^{n_w} \end{pmatrix} \quad (13)$$

$$\theta^T = ([\theta^1]^T, \dots, [\theta^{n_w}]^T)^T \quad (14)$$

$$\mathbf{s} = (\mathbf{s}^1, \dots, \mathbf{s}^{n_w})^T \quad (15)$$

\mathbf{L} is called the Locomotion matrix, analogous to the grasp matrix \mathbf{G} defined in [16], and \mathbf{J} is the Jacobian matrix.

2.2 Extension to arbitrary kinematics

In the previous section, the differential kinematic model was established for a fully parallel kinematics composed by several serial chains going from a platform to the wheel-ground contact. However, most actual rovers have a more complex structure : (i) the structure has branched kinematic chains, (ii) they can incorporate closed loops especially for actuation (pantographic mechanism), and (iii) they can use a coupler between two joints (like for the differential gear box used in the rocker-bogie system). This can be done simply by adding to the previous fully parallel model a set of algebraic equations expressing constraints between joint parameters θ^i

$$\Phi(\theta) = \mathbf{0} \quad (16)$$

By differentiating this equation with respect to time, we have

$$\Phi_\theta \dot{\theta} = \mathbf{0} \quad (17)$$

where $\Phi_\theta = [\frac{\partial \Phi}{\partial \theta}]$ is the (n_Φ, n_θ) Jacobian matrix of the algebraic constraint equation Φ . Bringing together the differential model (11) of the fully parallel kinematics and the latter constraint, the general form of an arbitrary rover kinematics can be written as

$$\begin{pmatrix} \mathbf{L}^T \\ \mathbf{0}_{n_\Phi, 6} \end{pmatrix} \mathbf{v}_p + \begin{pmatrix} \mathbf{J} \\ \Phi_\theta \end{pmatrix} \dot{\theta} = \begin{pmatrix} \mathbf{s} \\ \mathbf{0}_{n_\Phi, n_w} \end{pmatrix} \quad (18)$$

2.3 Dual static model

Let $\mathbf{f} = (f_x^1, f_y^1, f_z^1, \dots, f_x^{n_w}, f_y^{n_w}, f_z^{n_w})^T$ be the vector of contact force components, τ be a n_θ vector of actuator torques (and possibly all other non-conservative forces), and $\mathbf{w}_x, \mathbf{w}_\theta$ be generalized forces due to gravity (and possibly due to other conservative forces as elastic force exerted in suspension elements). They can be obtained from the potential energy function $U(\mathbf{x}, \theta)$

$$\mathbf{w}_x = -[\frac{\partial U}{\partial \mathbf{x}}] \quad (19)$$

$$\mathbf{w}_\theta = -[\frac{\partial U}{\partial \theta}] \quad (20)$$

The total virtual work developed by this set of forces can be written as

$$\begin{aligned} \delta W &= \delta \mathbf{x}_p^T \mathbf{w}_x + \delta \theta^T \tau + \delta \theta^T \mathbf{w}_\theta (\mathbf{L}^T \mathbf{A}_{pg} \delta \mathbf{x}_p + \mathbf{J} \delta \theta)^T \mathbf{f} \\ &= \delta \mathbf{x}_p^T (\mathbf{A}_{pg}^T \mathbf{L} \mathbf{f} + \mathbf{w}_x) + \delta \theta^T (\mathbf{J}^T \mathbf{f} + \tau + \mathbf{w}_\theta) \end{aligned}$$

The virtual displacement $\delta\theta$ must satisfy $\Phi_\theta\delta\theta = \mathbf{0}$, resulting from the holonomic constraint given by (16). Based on this, result and applying the principle of virtual works to the system defined by the generalized parameters $(\mathbf{x}_p, \theta)^T$, the equilibrium static equations can be written as

$$\begin{cases} \mathbf{A}_{pg}^T \mathbf{L} \mathbf{f} & = -\mathbf{w}_x \\ \mathbf{J}^T \mathbf{f} + \Phi_\theta^T \mathbf{m}_\phi & = -\boldsymbol{\tau} - \mathbf{w}_\theta \end{cases} \quad (21)$$

where \mathbf{m}_ϕ is a n_ϕ vector representing generalized forces (Lagrange multipliers) associated to constraints Φ .

The first group of this equation system characterizes the global equilibrium of the rover corresponding to generalized equations associated to \mathbf{x}_p . The second one deals with the internal forces associated to joint parameters θ .

As for grasping systems [22], kineto-static duality in locomotion systems can be represented on the diagram shown in Figure 2, using transformation matrices between joint, contact and task frames in velocity and force domains.

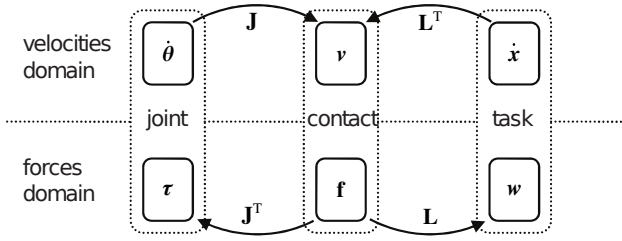


Fig. 2. Transformation matrices between joint, contact and task frames in both velocity and force domains.

2.4 Eliminating the steering rate and torque

Each existing joint in the rover kinematics has a dedicated role. It is

- either for suspension, creating a contact velocity along the contact normal direction z ,
- either for rolling, creating a contact velocity along the longitudinal direction x ,
- or either for steering, reorienting the wheel direction.

Then we can split the set of joint parameters θ into three sub-sets θ_s , θ_r and θ_t , for suspension, rolling and steering respectively. If the steering joint axis passes through the contact point (this is the case of most wheeled off-road vehicles) the Jacobian matrix is singular. The associated column to this steering joints is null or quasi-null, when the rover is moving on a flat surface or uneven surface, respectively. To avoid ill-conditionings of the Jacobian matrix, the steering

joints column and the steering rates $\dot{\theta}_t$ are removed, whilst considering the steering angle θ_t' in the calculation of other Jacobian columns, especially the ones associated to rolling joints. In the same manner, the moment equation giving the steering torque is removed in the static model.

2.5 Extension to dynamic model

The previous static equation can be extended in order to consider inertial forces. Motion equations derived from the Lagrangian formulation take the following form

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \left(\dot{\mathbf{M}}(\mathbf{q})\dot{\mathbf{q}} - \frac{\partial T}{\partial \dot{\mathbf{q}}} \right) + \begin{pmatrix} -\mathbf{w}_x \\ -\mathbf{w}_\theta \end{pmatrix} = \dots \\ \begin{pmatrix} \mathbf{0} \\ \boldsymbol{\tau} \end{pmatrix} + \begin{pmatrix} \mathbf{A}_{pg}^T \mathbf{L} \mathbf{0}_{6, n_\phi} \\ \Phi_\theta^T \end{pmatrix} \begin{pmatrix} \mathbf{f} \\ \mathbf{m}_\phi \end{pmatrix} \quad (22)$$

where $\mathbf{M}(\mathbf{q})$ is the mass matrix defined from the total kinetic energy $T = \frac{1}{2} \dot{\mathbf{q}}^T \mathbf{M}(\mathbf{q}) \dot{\mathbf{q}}$.

3 Contact friction model

A contact friction model is needed to complete and solve the previous mechanical model. Tire friction models developed for automotive applications as well as rigid wheel model used for planetary rover analysis have some common properties : they are generally given by an explicit function between tangential forces f_x, f_y and some normalized slippage ratios. These functions are dependent on the normal force f_z and other rheological and tribological parameters such as: tire stiffnesses; rubber-road friction coefficient and internal friction angle; cohesion and shearing modulus of granular soils. It is demonstrated from the brush model [23] that in the case of isotropic friction and combined slips, the total tangential force $\mathbf{f}_{xy} = (f_x, f_y)^T$ acts in a direction opposite to $\mathbf{s}_{xy} = (s_x, s_y)^T$, and its magnitude increases with the slip until reaching the maximal friction force μf_z (Fig.3). The same remark could be done for Bekker-Wong models [1, 2] and the rigid-wheel/terrain interaction. If we assume low slip level, we can use the decoupled linear friction model

$$\begin{cases} f_x = -c_x f_z s_k \\ f_y = -c_y f_z s_\alpha \end{cases} \quad (23)$$

where c_x and c_y are apparent contact stiffnesses, which do not only depend on the material stiffness of wheel and ground but also on the friction coefficient. s_k and s_α are the longitudinal slip ratio and the lateral sideslip defined by

$$\begin{cases} s_k = \frac{s_x}{v_x} \\ s_\alpha = \arctan \frac{v_y}{v_x} \simeq \frac{v_y}{v_x} \end{cases} \quad (24)$$

v_x and v_y are ground-velocity components of the wheel center. Assuming a negligible camber rate, $v_y \simeq s_y$, and a con-

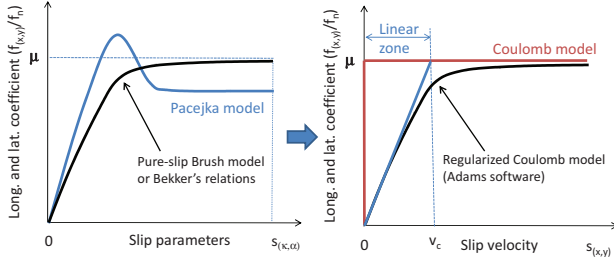


Fig. 3. Friction models comparison.

stant velocity displacement v_x , the contact model can be expressed by a linear function of slip velocities s_x and s_y

$$\begin{cases} f_x = -k_x f_z s_x \\ f_y = -k_y f_z s_y \end{cases} \quad (25)$$

This model approaches the friction model used in *Adams* software, in its linear zone (Fig. 3). This model is called sometimes as the regularized Coulomb model.

4 Solving the forward motion problem

The forward problem consists in determining the robot velocity for given actuator control inputs. We consider in our case that drive actuators are controlled in velocity, and steering ones are controlled in position. The control inputs are computed from a simple model based on planar kinematic motion and ideal rolling assumption (Appendix C). We also suppose that the suspension joints are passive, i.e. can be free or equipped with spring-damper devices.

The formalism presented previously allows the definition of the locomotion and Jacobian matrices. It is interesting to split equation (11) into two sets: the first one concerns the normal components along \mathbf{z} and the second one the tangential ones along \mathbf{x} and \mathbf{y} . This can be done by multiplying equation (11) by an appropriate selection matrices. We remind that these equations characterize the contact velocity expressed in the local contact frame. It follows:

$$\begin{cases} \mathbf{s}_z = \mathbf{L}_z^T \mathbf{v}_p + \mathbf{J}_z \dot{\boldsymbol{\theta}} \\ \mathbf{s}_{xy} = \mathbf{L}_{xy}^T \mathbf{v}_p + \mathbf{J}_{xy} \dot{\boldsymbol{\theta}} \end{cases} \quad (26)$$

where \mathbf{L}_z and \mathbf{J}_z are locomotion and Jacobian matrices \mathbf{L} and \mathbf{J} , restricted to the z normal components; \mathbf{L}_{xy} and \mathbf{J}_{xy} their restriction to the x, y tangential components. $\mathbf{s}_z = (s_z^1, \dots, s_z^n)^T$ is the vector of normal velocities, and $\mathbf{s}_{xy} = (s_x^1, s_y^1, \dots, s_x^n, s_y^n)^T$ is the vector of slip velocities. We assume that the contact is permanent and continuous, and therefore \mathbf{s}_z may be considered equal to zero. However, this only forms a

necessary condition but not sufficient to maintain contact because of time-integration errors. The stabilization of the contact constraints can be then achieved by considering a normal contact velocity vector \mathbf{s}_z in (26), computed from the distance d_{cg} between the surfaces of the wheel and the ground, such as

$$\mathbf{s}_z^i = -\eta_d d_{cg}^i = -\eta_c (z_C^i - z_g^i) \quad (27)$$

where η_c and η_d are positive gains; z_C^i and z_g^i the heights of the two closest points on the wheel and the ground surface. At the same time, this correction term allows us to solve the initial kinematic problem, i.e. the whole-configuration of the rover at its starting position.

Considering this and splitting joint rate terms making appear suspension velocities and rolling ones $(\dot{\boldsymbol{\theta}}_s, \dot{\boldsymbol{\theta}}_r)$, we can write

$$\begin{cases} \mathbf{L}_z^T \mathbf{v}_p + \mathbf{J}_{z,s} \dot{\boldsymbol{\theta}}_s + \mathbf{J}_{z,r} \dot{\boldsymbol{\theta}}_r = \mathbf{s}_z \\ \mathbf{L}_{xy}^T \mathbf{v}_p + \mathbf{J}_{xy,s} \dot{\boldsymbol{\theta}}_s + \mathbf{J}_{xy,r} \dot{\boldsymbol{\theta}}_r = \mathbf{s}_{xy} \end{cases} \quad (28)$$

The wheel's rotation does not create a velocity component along the contact normal direction, hence $\mathbf{J}_{z,r} = \mathbf{0}$. $\dot{\boldsymbol{\theta}}_r$ is the control inputs and is computed here from a simple planar model (Appendix C).

Let's call the $2n \times 2n$ diagonal matrix $\mathbf{K}(\mathbf{f}_z) = \text{diag}(k_x^1 f_z^1, k_y^1 f_z^1, k_x^2 f_z^2, k_y^2 f_z^2, \dots, k_x^n f_z^n, k_y^n f_z^n)$ representing the contact stiffness matrix. Substituting now the linear contact model (25), written in its matrix form for all contacts,

$$\mathbf{f}_{xy} = -\mathbf{K}(\mathbf{f}_z) \mathbf{s}_{xy} \quad (29)$$

into the equilibrium equation (21), we obtain

$$\begin{cases} \mathbf{L}_z \mathbf{f}_z - \mathbf{L}_{xy} \mathbf{K}(\mathbf{f}_z) \mathbf{s}_{xy} = -\mathbf{A}_{pg}^{-T} \mathbf{w}_x \\ \mathbf{J}_{z,s}^T \mathbf{f}_z - \mathbf{J}_{xy,s}^T \mathbf{K}(\mathbf{f}_z) \mathbf{s}_{xy} + \Phi_{\boldsymbol{\theta}_s}^T \mathbf{m}_\phi = -\boldsymbol{\tau}_s - \mathbf{w}_s \\ \mathbf{J}_{z,r}^T \mathbf{f}_z - \mathbf{J}_{xy,r}^T \mathbf{K}(\mathbf{f}_z) \mathbf{s}_{xy} = -\boldsymbol{\tau}_r - \mathbf{w}_r \end{cases} \quad (30)$$

The second and the third relations are equilibrium equations associated to the suspension joint parameters $\boldsymbol{\theta}_s$ and the rolling joint parameters $\boldsymbol{\theta}_r$, respectively. We consider that wheel drive actuators are independent ($\boldsymbol{\theta}_r$ are independent), hence $\Phi(\boldsymbol{\theta}) = \Phi(\boldsymbol{\theta}_s)$. $\mathbf{w}_{s,r}$ and $\boldsymbol{\tau}_{s,r}$ are conservative and non-conservative generalized forces associated to $\boldsymbol{\theta}_s$ and $\boldsymbol{\theta}_r$. The center of gravity of the wheels are located on their axles, hence $\mathbf{w}_r = \mathbf{0}$. The last equation of (30) forms n scalar equations and introduces n unknown wheel torques $\boldsymbol{\tau}_r$; these latter are not included in the two other equations; then this equation will be ignored for the moment.

By substituting \mathbf{s}_{xy} from (28) into (30) and considering the no-separation constraint (28) and joint constraints (17),

the whole-body mechanical system can be described by the equation system

$$\begin{cases} \mathbf{L}_z \mathbf{f}_z - \mathbf{L}_{xy} \mathbf{K}(\mathbf{f}_z) (\mathbf{L}_{xy}^T \mathbf{v}_p + \mathbf{J}_{xy,s} \dot{\theta}_s + \mathbf{J}_{xy,r} \dot{\theta}_r) = -\mathbf{A}_{pg}^{-T} \mathbf{w}_x \\ \mathbf{J}_{z,s}^T \mathbf{f}_z - \mathbf{J}_{xy,s}^T \mathbf{K}(\mathbf{f}_z) (\mathbf{L}_{xy}^T \mathbf{v}_p + \mathbf{J}_{xy,s} \dot{\theta}_s + \mathbf{J}_{xy,r} \dot{\theta}_r) + \Phi_{\theta_s}^T \mathbf{m}_\phi \dots \\ \hspace{15em} = -\boldsymbol{\tau}_s - \mathbf{w}_s \\ \mathbf{L}_z^T \mathbf{v}_p + \mathbf{J}_{z,s} \dot{\theta}_s = \mathbf{s}_z \\ \Phi_{\theta_s} \dot{\theta}_s = \mathbf{0} \end{cases}$$

The forward problem consists in determining the robot velocity ($\mathbf{v}_p, \dot{\theta}_s$) and static components (\mathbf{f}_z) for a given actuator control inputs composed by wheel rates $\dot{\theta}_r$ and steering angles θ_r . It is easy to demonstrate that this system equation is a square non-linear system of dimension $2n + m + 3$. It is supposed here that the suspension system has no redundancy and the dimension of independent suspension parameters is equal to $n - 3$.

Nonlinear terms which represent tangential force components can be linearized by using an estimated value of \mathbf{f}_z . For this, we will compute the least square solution $\bar{\mathbf{f}}$ of contact force using the static model (21) restricted to equilibrium equation associated to generalized parameter \mathbf{x} and θ_s . By introducing \mathbf{B} , the null space matrix of Φ_{θ_s} , these latter equilibrium equations can be written as

$$\begin{pmatrix} \mathbf{L} \\ \mathbf{B}^T \mathbf{J}_s^T \end{pmatrix} \mathbf{f} = \begin{pmatrix} -\mathbf{A}_{pg}^{-T} \mathbf{w}_x \\ -\mathbf{B}^T \mathbf{w}_s \end{pmatrix} \quad (31)$$

The restriction to suspension joint parameters and the multiplication by \mathbf{B}^T allow to eliminate the actuator torques $\boldsymbol{\tau}_r$ and Lagrange multipliers \mathbf{m}_ϕ , thus providing an homogeneous unknown vector \mathbf{f} . Then, the least square solution $\bar{\mathbf{f}}$ of this underdetermined system gives an estimated value of contact normal force $\bar{\mathbf{f}}_z$ which has a certain physical meaning. Using the estimated value of contact stiffness $\mathbf{K}(\bar{\mathbf{f}}_z)$, equation (31) yields to the linear system

$$\begin{pmatrix} \mathbf{L}_{xy} \mathbf{K}(\bar{\mathbf{f}}_z) \mathbf{L}_{xy}^T & \mathbf{L}_{xy} \mathbf{K}(\bar{\mathbf{f}}_z) \mathbf{J}_{xy,s} & -\mathbf{L}_z & \mathbf{0}_{6,m} \\ \mathbf{J}_{xy,s}^T \mathbf{K}(\bar{\mathbf{f}}_z) \mathbf{L}_{xy}^T & \mathbf{J}_{xy,s}^T \mathbf{K}(\bar{\mathbf{f}}_z) \mathbf{J}_{xy,s} & -\mathbf{J}_{z,s}^T & -\Phi_{\theta_s}^T \\ -\mathbf{L}_z^T & -\mathbf{J}_{z,s} & \mathbf{0}_{n,n} & \mathbf{0}_{n,m} \\ \mathbf{0}_{m,6} & -\Phi_{\theta_s} & \mathbf{0}_{m,n} & \mathbf{0}_{m,m} \end{pmatrix} \begin{pmatrix} \mathbf{v}_p \\ \dot{\theta}_s \\ \mathbf{f}_z \\ \mathbf{m}_\phi \end{pmatrix} \dots \\ = \begin{pmatrix} \mathbf{A}_{pg}^{-T} \mathbf{w}_x - \mathbf{L}_{xy} \mathbf{K}(\bar{\mathbf{f}}_z) \mathbf{J}_{xy,r} \dot{\theta}_r \\ \mathbf{w}_s - \mathbf{J}_{xy,s}^T \mathbf{K}(\bar{\mathbf{f}}_z) \mathbf{J}_{xy,r} \dot{\theta}_r \\ -\mathbf{s}_z \\ \mathbf{0}_{m,1} \end{pmatrix} \quad (32)$$

The matrix on the left-hand side is symmetrical. It turns out from numerical test that it is also a definite positive matrix, and hence the Cholesky method can be applied to solve this system.

5 Initial pose computation

The differential kinematic method requires the knowledge of the initial configuration of the rover, i.e. its position, posture and suspension joint positions. The initial pose

computation is obvious when the local ground surface is flat. However it is not easy to compute when the ground surface is rough since this problem is a non-linear kinematic problem and generally solved by the well known iterative Newton-Raphson method, which is based on the Jacobian matrix of the constraint equations. Our method is quite similar but takes advantage of the differential kinematic model described in Section 2. The idea of the approach is to start at an arbitrary rover configuration over the ground surface and "to send down" until all wheels touch the surface, as illustrated in (Fig. 4). This can be done by controlling the normal velocity in the wheel-ground contact by the function

$$s_z^i = -k_z(z_C^i - z_g^i) \quad (33)$$

where z_C^i and z_g^i are the altitudes of the two closest points on wheel i and on the ground surface.

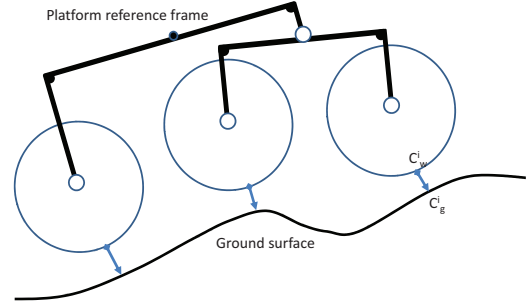


Fig. 4. Iterative process for computing the initial complete robot pose.

Tangential non-slip contact equations are relaxed in this process. If we consider only the velocity equations along the contact normals and only the suspension joint rates, equation (9) for the i th contact becomes:

$$\mathbf{L}_z^T \mathbf{A}_{pg} \dot{\mathbf{x}}_p + \mathbf{J}_{z,s} \dot{\theta}_s = \mathbf{s}_z \quad (34)$$

This equation, associated with the constraint Jacobian equation (17), yields to

$$\begin{pmatrix} \mathbf{L}_z^T \mathbf{A}_{pg} \bar{\mathbf{A}}_o & \mathbf{J}_{z,s} \\ \mathbf{0}_{n\phi,3} & \Phi_{n\phi, n\theta_s} \end{pmatrix} \begin{pmatrix} \dot{\mathbf{x}}_{po} \\ \dot{\theta}_s \end{pmatrix} = \begin{pmatrix} \mathbf{s}_z \\ \mathbf{0}_{n\phi} \end{pmatrix} \quad (35)$$

where $\bar{\mathbf{A}}_o$ is the selection matrix which selects the elevation, roll and pitch components in the platform twist $\dot{\mathbf{x}}_{po} = (\dot{z}, \dot{o}_x, \dot{o}_y)^T$. The other platform twist components are equal to zero constraining the x and y coordinates of the platform point and its yaw angle o_z to have constant values defined by the user. Rolling rates are not considered because

they introduce here a redundancy for this problem and steering joint rates are eliminated for the same reasons mentioned previously in Section 2.4. However, the steering joint angles are actually considered in this model when computing the local contact frame \mathcal{R}_c . For the system suspension with a minimal degree of freedom, the latter linear system equation is a determined square system of dimension $n_w + n_\phi$, and gives a unique solution composed by the 3 components of $\dot{\mathbf{x}}_{po}$ and the $n_w - 3$ components of $\dot{\theta}$). A pseudo-time integration of the rover parameters allows us to find a new configuration where distances between the wheels and the ground surface are smaller. The process is repeated until these distances drop below a predefined threshold.

6 Application to the rocker-bogie system

The developed models will be applied to the motion simulation of the well-known rocker-bogie chassis [24]. First, we will present the rover kinematics and define its parametrization. This system has six driven wheels, the only front and rear wheels are steered. On each side, front and central wheels are mounted on a bogie which is connected to a rocker by a passive rotoide joint. The two rockers are articulated to the platform by two passive rotoide joints which rotate oppositely by means of a differential gearbox. Figures 5 and 6 give the rover kinematics, kinematic chain numbers, joint parameters and structural parameters.

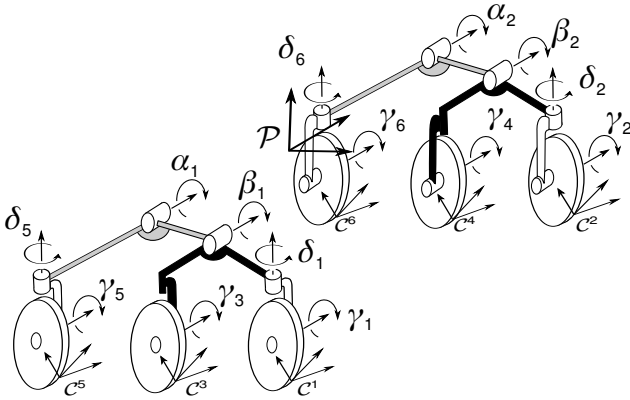


Fig. 5. Kinematics of a rocker-bogie rover and joint parameters.

As explained in the previous section, the branched kinematic chains of the rover can be first considered as a fully parallel kinematics forming a star graph, and afterwards algebraic constraints between joint parameters can be added (expressing that a joint parameter in chain i is equal to that in the chain j). This permits in all cases, to write easily the block-diagonal Jacobian matrix. For the particular case of Rocker-bogie system and for obvious reasons of computing efficiency, we will give here the Jacobian matrix of the actual kinematics with a minimal set of joint parameters. The joint

parameters, defined in Figure 5, are ordered as follow:

$$\theta = (\alpha_1, \alpha_2, \beta_1, \beta_2, \gamma_1, \gamma_2, \gamma_3, \gamma_4, \gamma_5, \gamma_6)^T \quad (36)$$

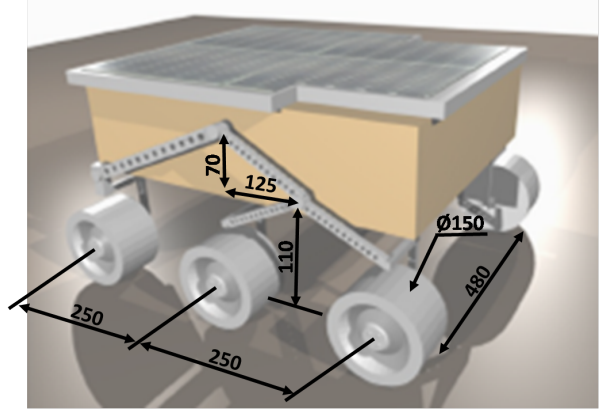


Fig. 6. Kinematic dimensions of the rocker-bogie robot.

The Jacobian matrix is given by

$$\mathbf{J} = \begin{pmatrix} \mathbf{j}_{\alpha_1}^1 & \mathbf{0} & \mathbf{j}_{\beta_1}^1 & \mathbf{0} & \mathbf{j}_{\gamma_1}^1 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{j}_{\alpha_2}^2 & \mathbf{0} & \mathbf{j}_{\beta_2}^2 & \mathbf{0} & \mathbf{j}_{\gamma_2}^2 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{j}_{\alpha_1}^3 & \mathbf{0} & \mathbf{j}_{\beta_1}^3 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{j}_{\gamma_3}^3 & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{j}_{\alpha_2}^4 & \mathbf{0} & \mathbf{j}_{\beta_2}^4 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{j}_{\gamma_4}^4 & \mathbf{0} & \mathbf{0} \\ \mathbf{j}_{\alpha_1}^5 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{j}_{\gamma_5}^5 & \mathbf{0} \\ \mathbf{0} & \mathbf{j}_{\alpha_2}^6 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{j}_{\gamma_6}^6 \end{pmatrix}$$

where $\mathbf{0}$ are a 3D zero column vector of size 3, and $\mathbf{j}_{\theta}^i = (\mathbf{e}_\theta \times \mathbf{r}_{P\theta})_{\mathcal{R}_c}^i$ with \mathbf{e}_θ is the unit vector of the joint axis, P_θ an arbitrary point of this axis, i corresponding to the index of the contact point. We notice that for all i $\mathbf{j}_{\gamma_i}^i = (-r, 0, 0)^T$ with r the wheel radius.

The main geometrical dimensions of the articulated rover are given in Figure (6).

The locomotion matrix of the i th wheel is demonstrated to be equal

$$\mathbf{L}^i = (\mathbf{R}_{cp}^i \quad \hat{\mathbf{u}}_{cp}^i \mathbf{R}_{cp}^i) \quad (37)$$

where \mathbf{R}_{cp}^i is the matrix rotation from the platform frame \mathcal{P} to the contact one \mathcal{C}^i and $\hat{\mathbf{u}}_{cp}^i$ is the skew-symmetric matrix of vector cross-product, with \mathbf{u}_{cp}^i the position vector of the platform center expressed in the contact coordinate system.

The constraint introduced by the differential gearbox coupling the two rockers can be written by

$$\Phi(\theta) = \Phi(\theta_s) \equiv \alpha_1 + \alpha_2 = 0 \quad (38)$$

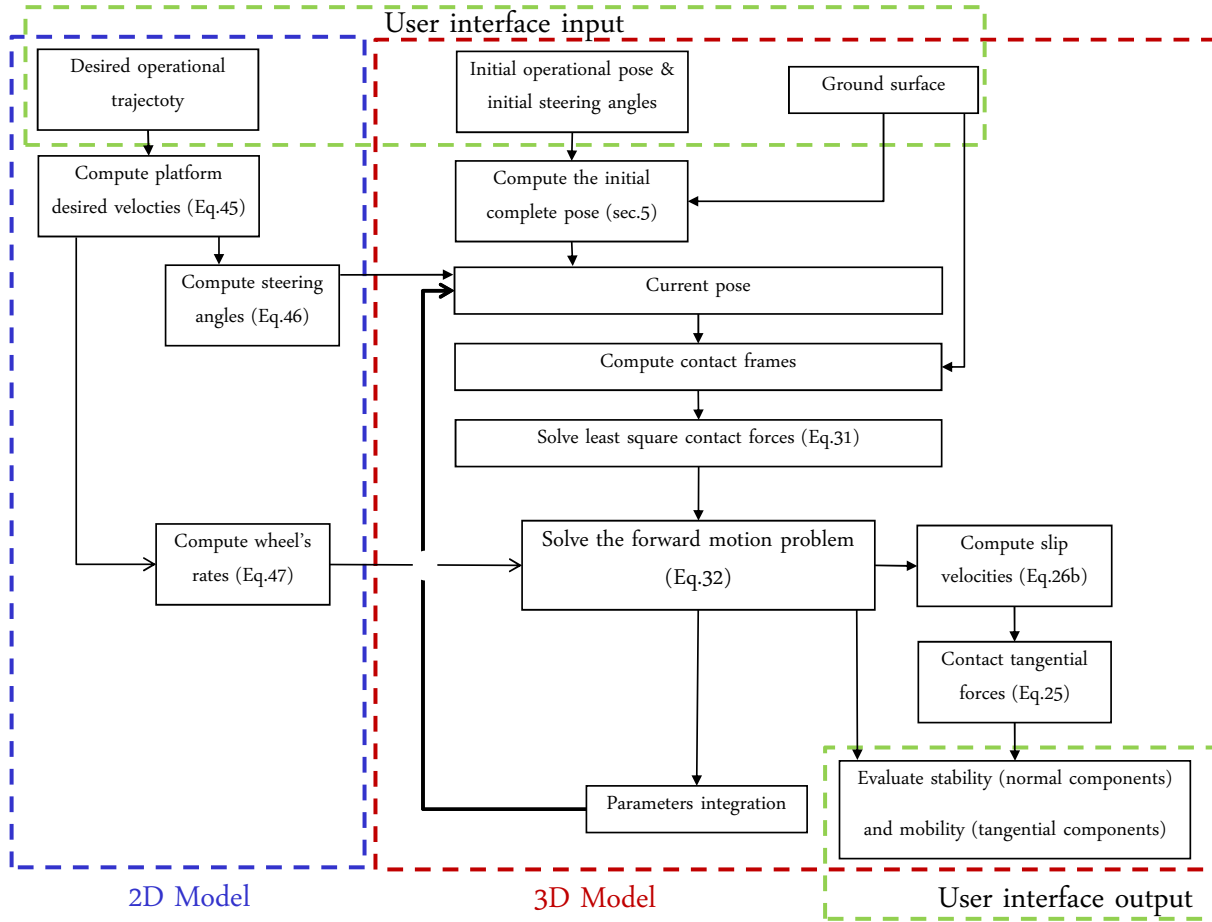


Fig. 7. Simulation scheme of rover quasi-static motion.

and the associated Jacobian

$$\Phi_{\theta_s} = (1, 1, 0, 0) \quad (39)$$

The general architecture of the simulation framework is summarized in Figure (7). The method integrates velocity components with respect to time in order to obtain the configuration parameters, assuming that the initial configuration is known. For given x, y coordinates of the platform point P and a given yaw angle, the other configuration parameters including the platform posture and suspension joint angles are computed by the initial pose function. The physical evaluation of the pose is done once the distribution of the contact forces is solved. In general, the stability and the mobility of the rover are considered to be satisfied if each normal component is positive and each total tangential force is inside the friction cone. As said before, this framework can be used for on-line and off-line path planning, which assumes the knowledge of the ground surface conditions over a given horizon, and especially its surface map. For our model, the map can be represented by any continuous and differentiable C_1 func-

tion, where the contact normal and tangential plane can be computed at each surface point. For the actual application, this map can be measured by exteroceptive sensors, generally by a Lidar or a stereo camera.

7 Simulation results

This section presents simulation results illustrating quasi-static rover motion when crossing different ground surfaces. The previous kineto-static model is solved using Matlab software, and the geometrical parameters are integrated with respect to time thanks to the available ODE functions. These results are compared to other simulation results given by the Adams dynamic software. The latter computes full dynamics of the rover and uses a regularized non-linear Coulomb law for modeling friction in wheel-ground contacts, as explained in Section 3. Also, the normal contact force is computed by a penalty method which consider the interpenetration of rigid surfaces and a virtual spring-damper model. The goals of this comparison are (i) to validate the kinematic results of our simulation model and (ii) to question about the validity

of the quasi-static assumption. In the two simulators, the wheel velocities are equal and defined such that the desired velocity displacement is equal $0.1m/s$. For all the figures in this section, the red curves represent results given by the kineto-static model and the blue ones for the dynamic model.

7.1 Crossing an asymmetrical obstacle

In this first example, the robot has to cross in straight line two extruded surfaces, for right and left wheels. Both extruded surfaces are defined by a sigmoid function and are shifted the one with respect to the other (see Figure 8). Since the horizontal set-point path is a straight line, the steering angles are zeros and the wheels rates are identical. This example is chosen deliberately simple to permit comparison between the two models.

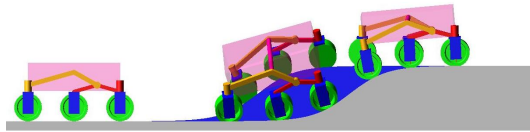


Fig. 8. Adams GUI : crossing two shifted sigmoids.

Geometrical parameters are plotted in Figure 9 as function of the longitudinal displacement abscissa. These parameters include roll and pitch angles of the platform, its center of gravity height, the right rocker angle, the left and right bogie angles, the horizontal performed path and the yaw angle. Some differences between kineto-static results and the Adams dynamic ones are observed as for the roll and pitch angles. We think these differences can be explained by the difference of wheel geometries between the two models (a disk for the first one and a cylinder for the second). However, the major relevant result concerns the horizontal path and the yaw angle, demonstrating the validity of the model to handle the sideslip phenomenon and path deviation. Lateral slip velocities are compared in Figure 10 which confirms the previous observation. These slippages are due to the asymmetry of the ground features between right and left tangent planes.

Longitudinal slips, given in Figure 11, appear to match closely to the one given by Adams. These slips are due to the ground roughness and to the incompatibility of wheel velocities, assumed here to be equal. We mention that this kind of control, based on a 2D differential kinematic model, is a usual way to determine the velocity inputs. Next, the actual contact normals are difficult to measure directly or to estimate accurately for a rover when moving on a rough surface.

Tangential (longitudinal and lateral) force curves look

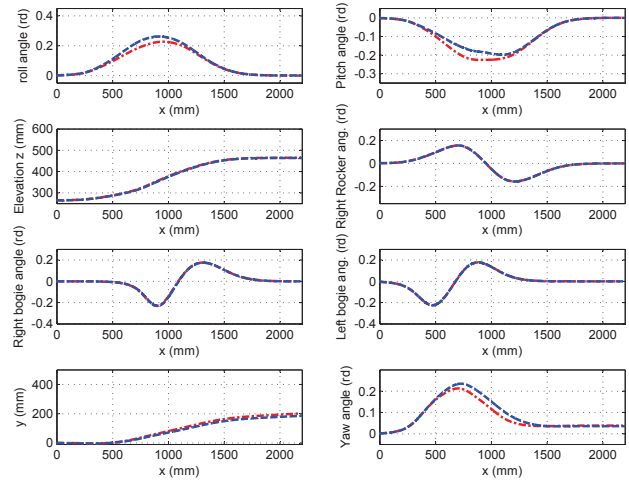


Fig. 9. Configuration parameters as function of x abscissa : kineto-static model (red) vs. dynamic model (blue).

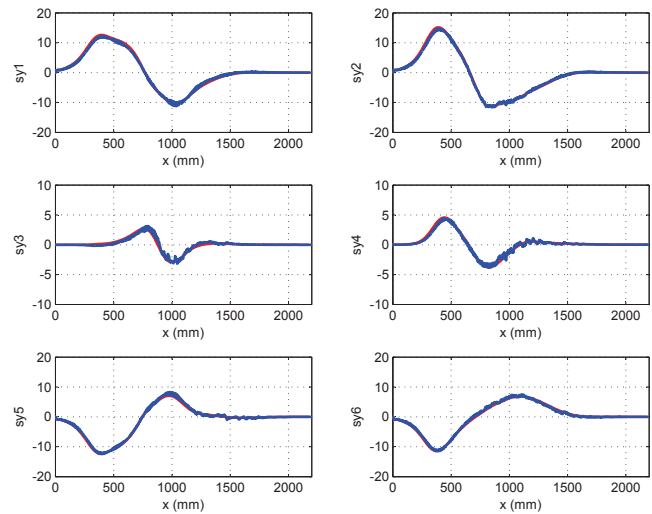


Fig. 10. Lateral slip velocity as function of x abscissa : kineto-static model (red) vs. dynamic model (blue).

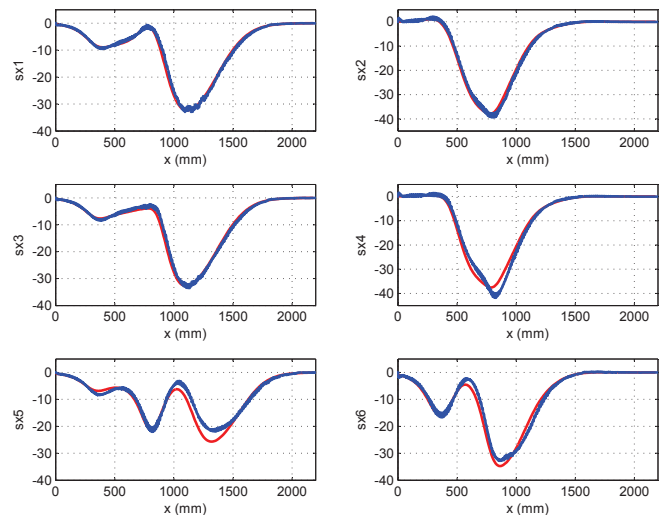


Fig. 11. Longitudinal slip velocity as function of x abscissa : kineto-static model (red) vs. dynamic model (blue).

similar to their corresponding slip velocities since the first ones are proportional and opposite to the second ones. Moreover, the normal forces given in Figure 12, show a good agreement and demonstrate the validity of the prediction-correction method allowing the resolution of the nonlinear kineto-static model (30). These forces allow the stability evaluation of the robot along the path; the minimal force value could be used as a stability margin.

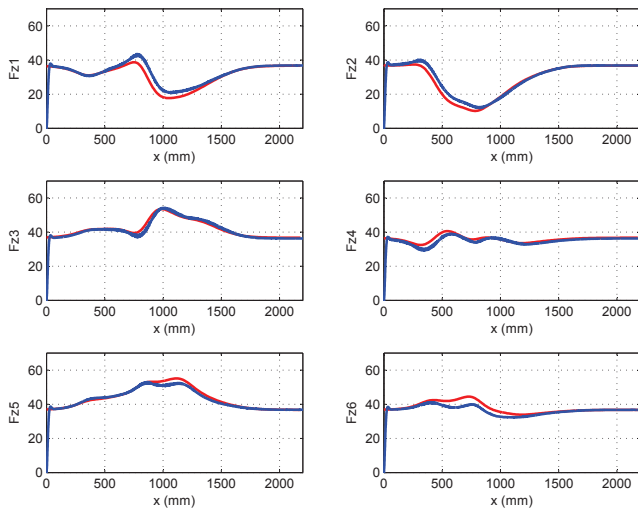


Fig. 12. Normal forces as function of x abscissa : kineto-static model (red) vs. dynamic model (blue).

7.2 Crossing diagonally a corrugated surface

In this second example, the robot has to move diagonally in straight line on a corrugated surface defined by an extrusion of a sinusoid function (Figure 13). The robot starts at the origin point and the prescribed path is a straight line. The control inputs are similar to that used in the previous case.

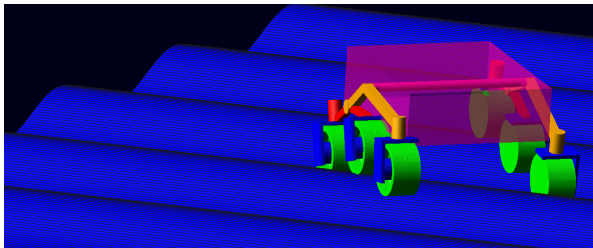


Fig. 13. Adams GUI : crossing diagonally a corrugated surface.

Figure 14 gives performed paths and the yaw angles computed by the two models. We observe high path deviations with respect to the prescribed one (square's diagonal) and important oscillations of the robot yaw angles. The results of the proposed model match well with that computed

by Adams and demonstrate the validity of our model to predict the robot motion and slippages. The curve differences between the two models increase with time and traveled distance. This is due to the fact that the wheels in both models don't follow exactly the same ground tracks thus limiting the comparison to short term travel. Figure 15 plots the horizontal coordinates of wheels traces. The black line represents the robot center path. The robot motion tends to align its configuration with that of the extrusion, minimizing thus the contact-slip level and the energy consumption.

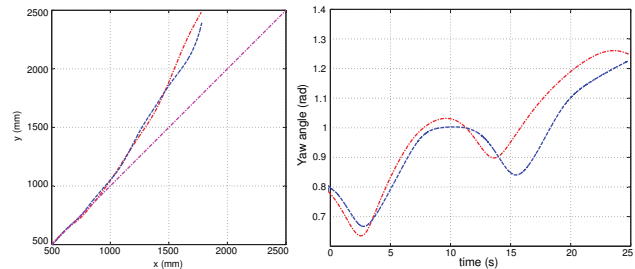


Fig. 14. Performed path and yaw angles : kineto-static model (red) vs. dynamic model (blue).

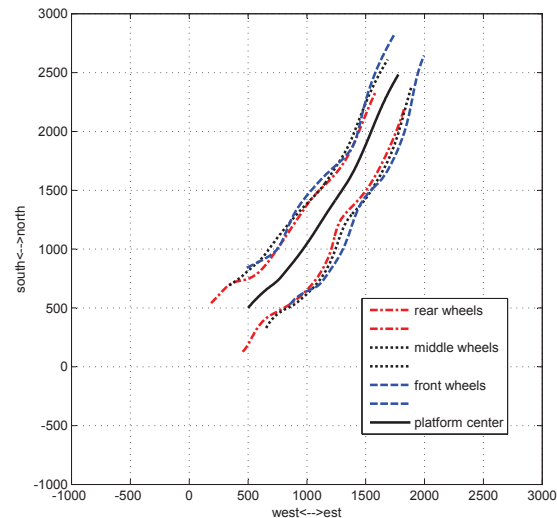


Fig. 15. Wheel's traces.

7.3 A complex rough surface

The surface is defined here by a cubic spline approximating a set of points defined on a planar grid. The point heights are generated by a fractal method based on a recursive subdivision of the terrain height map. At each step of the subdivision, new height values are calculated by taking the average of the neighboring height values and

applying a random offset.

Figure 16 shows the ground surface and plots the trajectories of middle wheel centers computed by the kineto-static model and by the dynamic one. The robot is supposed to travel in straight line along the diagonal. The top view image shows a good agreement of the computed paths proving the validity of the proposed model to predict lateral slip and path deviation. Some large differences of the computed paths along the vertical direction can be observed on the 3D viewpoint. This is due to the difference of the wheel geometry between the two models (a disk for the kineto-static model and a cylinder for the dynamic one).

8 Conclusion

Quasi-static slipping motion of articulated rovers can be approached by a coupled model, including the differential kinematic one, the static one and a wheel-ground contact model. Assuming small slip rates, this latter can be approximated by a linear relationship between forces and velocities. The proposed model approaches the system dynamics since it predicts with good accuracy and low computation time, contact slips, contact forces and the robot trajectory. The method is sufficiently generic to consider most of rover kinematics having articulated structure with coupled or independent suspensions. The model is simple enough and computationally efficient to be implemented on-board for autonomous task-planning and for path evaluation. Computation time for the presented examples is less than 1 second per meter path length using Matlab software. This can be drastically reduced if coded in any low level language. The developed model will be extended to consider basic concepts of terramechanics such as rolling and lateral resistance due to soil compaction and bulldozing effect. These components can be easily added as external forces acting on the wheel center. Finally, contact computation will be improved in our simulator to consider more complex wheel shapes and multiple contact points cases.

References

- [1] Bekker, M., 1969. *Introduction to terrain-vehicle systems*. Ann arbor, University of Michigan Press.
- [2] Wong, J., 1989. *Terramechanics and off-road Vehicles*. Elsevier.
- [3] Iagnemma, K., Kang, S., Shibly, H., and Dubowsky, S., 2004. "Online terrain parameter estimation for wheeled mobile robots with application to planetary rovers". *IEEE Transactions on Robotics and Automation*, **20**(5), pp. 921–927.
- [4] Jain, A., Guineau, J., Lim, C., Lincoln, W., Pomerantz, M., Sohl, G., and Steele, R., 2003. "Roams: Planetary surface rover simulation environment,". In Proceedings of the International Symposium on Artificial Intelligence Robotics and Automation in Space (i-SAIRAS).
- [5] Howard, T., and Kelly, A., 2007. "Optimal rough terrain trajectory generation for wheeled mobile robots". *International Journal of Robotics Research*, **26**(1), February, pp. 141–166.
- [6] Gibbesch, A., and Schafer, B., 2005. "Multibody system modelling and simulation of planetary rover mobility on soft terrain". In Proc. of The 8th International Symposium on Artificial Intelligence, Robotics and Automation in Space (iSAIRAS), Munich, Germany.
- [7] Thueer, T., Ambrose Krebs, A., Siegart, R., and Lamon, P., 2007. "Performance comparison of rough-terrain robots simulation and hardware". *Journal of Field Robotics*, **24**(3), pp. 251–271.
- [8] Ishigami, G., Nagatani, K., and Yoshida, K., 2007. "Path planning for planetary exploration rovers and its evaluation based on wheel slip dynamics". In Proc. IEEE Int Robotics and Automation Conf, pp. 2361–2366.
- [9] Andrade, G., Benamar, F., Bidaud, P., and Chatila, R., 1998. "Modeling robot-soil interaction for planetary rover motion control". In Proc. Conf. IEEE/RSJ Int Intelligent Robots and Systems, Vol. 1, pp. 576–581.
- [10] Krebs, A., Thueer, T., Michaud, S., and Siegart, R., 2006. "Performance optimization of all-terrain robots: A 2d quasi-static tool". In Proc. IEEE/RSJ Int Intelligent Robots and Systems Conf, pp. 4266–4271.
- [11] Benamar, F., Jarrault, P., Bidaud, P., and Grand, C., 2009. "Analysis and optimization of obstacle clearance of articulated rovers". In Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems IROS 2009, pp. 4128–4133.
- [12] Grand, C., Benamar, F., Plumet, F., and Bidaud, P., 2004. "Stability and traction optimisation of high mobility rover - application to hybrid wheel-leg rover". *The International Journal of Robotics Research*, **23**(10-11), pp. 1041–1058.
- [13] Sreenivasan, S., and Wilcox, B., 1994. "Stability and traction control of an actively actuated micro-rover". *Journal of Robotics Systems*, **11**(6), pp. 487–502.
- [14] Tarokh, M., and McDermott, G. J., 2005. "Kinematics modeling and analyses of articulated rovers". *IEEE Transactions on Robotics*, **21**(4), pp. 539–553.
- [15] Auchter, J., and Moore, C. A., 2008. "Off-road robot modeling with dextrous manipulation kinematics". In Proc. IEEE Int. Conf. Robotics and Automation ICRA 2008, pp. 2313–2318.
- [16] Li, Z., Hsu, P., and Sastry, S., 1989. "Grasping and coordinated manipulation by a multifingered robot hand". *Int. J. of Robotics Research*, **8**, (4), 8-4, pp. 33–50.
- [17] Chakraborty, N., and Ghosal, A., 2004. "Kinematics of wheeled mobile robots on uneven terrain". *Mechanism and Machine Theory*, **39**, pp. 1273–1287.
- [18] Trinkle, J., 1989. "A quasi-static analysis of dextrous manipulation with sliding and rolling contacts". In Robotics and Automation, 1989. Proceedings., 1989 IEEE International Conference on, pp. 788–793 vol.2.
- [19] Peshkin, M., and Sanderson, A., 1989. "Minimization of energy in quasi-static manipulation". *Robotics and Automation, IEEE Transactions on*, **5**(1), feb, pp. 53–60.

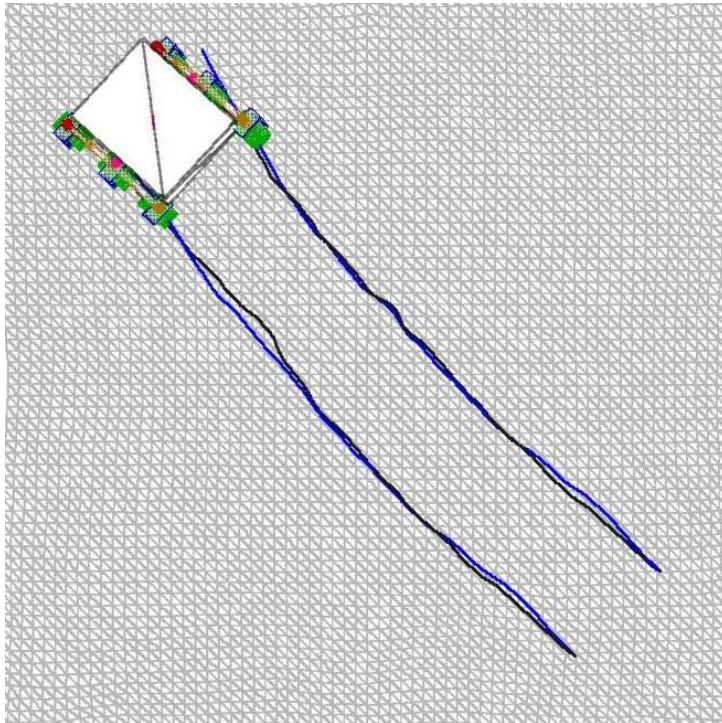
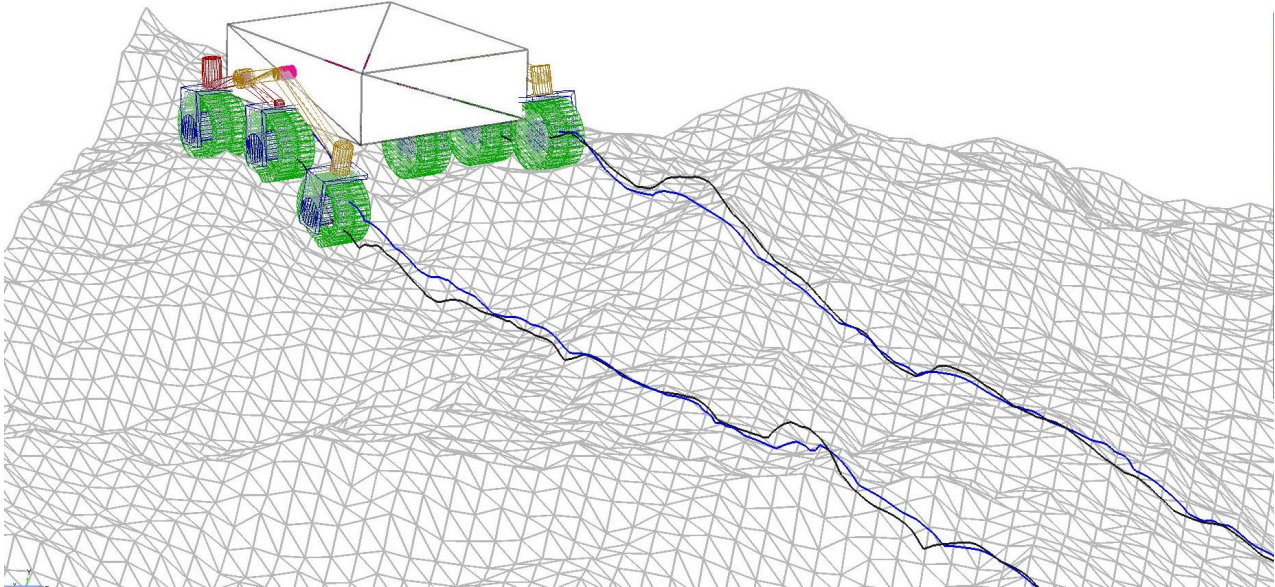


Fig. 16. Trajectory comparison of middle wheel centers when traveling on a highly rough surface.

- [20] Kumar, V., and Waldron, K., 1988. "Force distribution in closed kinematic chains". *IEEE Journal of Robotics and Automation*, **4**(6), pp. 657–663.
- [21] Selig, J., 2005. *Geometric fundamentals of robotics*. Springer, New York.
- [22] Murray, R., Li, Z., and Sastry, S., 1994. *A Mathematical Introduction to Robotic Manipulation*. CRC Press.
- [23] Pacejka, H., 2002. *Tyre and vehicle dynamics*. Butterworth-Heinemann.
- [24] D. Bickler, U., 1989. "Articulated suspension systems". *US Patent Office 4,840,394, Washington, D.C.*

A Notation

The twist regroups the linear and the angular velocity

$$\mathcal{S} = \begin{pmatrix} \mathbf{v} \\ \boldsymbol{\omega} \end{pmatrix} \quad (40)$$

The hat operator $\hat{\mathbf{a}}$ represents the skew-symmetric matrix associated to vector cross-product

$$\hat{\mathbf{a}} = \begin{pmatrix} 0 & -a_z & a_y \\ a_z & 0 & -a_x \\ -a_y & a_x & 0 \end{pmatrix} \quad (41)$$

B Roll, pitch and yaw rotations

Platform configurations relative to ground is parameterized by $\mathbf{x}_p = (x_p, y_p, z_p, o_x, o_y, o_z)^T$ which regroups the position of $O_p = (x_p, y_p, z_p)^T$ and the usual roll, pitch and yaw angles $(o_x, o_y, o_z)^T$. Those angles are defined so that \mathcal{G} turns first about the z -axis with angle o_z , then about the y -axis with angle o_y and finally about the x -axis with angle o_x . Then rotation matrix that transforms a point coordinate from \mathcal{P} to \mathcal{G} can be computed by

$$\mathbf{R}_{gp} = \begin{pmatrix} \text{Co}_z & -\text{So}_z & 0 \\ \text{So}_z & \text{Co}_z & 0 \\ 0 & 0 & 1 \end{pmatrix} \dots \begin{pmatrix} \text{Co}_y & 0 & \text{So}_y \\ 0 & 1 & 0 \\ -\text{So}_y & 0 & \text{Co}_y \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & \text{Co}_x & -\text{So}_x \\ 0 & \text{So}_x & \text{Co}_x \end{pmatrix} \quad (42)$$

The platform motion w.r.t. ground are given by the twist expressed in \mathcal{P}

$$\mathbf{v}_p = [\$_{gp}]_{O_p}^p = \begin{pmatrix} \mathbf{R}_{gp}^T & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & \mathbf{G}_o \end{pmatrix} \begin{pmatrix} \dot{x}_p \\ \dot{y}_p \\ \dot{z}_p \\ \dot{o}_x \\ \dot{o}_y \\ \dot{o}_z \end{pmatrix} = \mathbf{A}_{pg} \dot{\mathbf{x}}_p \quad (43)$$

with

$$\mathbf{G}_o = \begin{pmatrix} 1 & 0 & -\text{So}_y \\ 0 & \text{Co}_x & \text{So}_x \text{Co}_y \\ 0 & -\text{So}_x & \text{Co}_x \text{Co}_y \end{pmatrix} \quad (44)$$

C Inputs computation

Inputs parameters of the previous direct differential kinematic model are composed by:

- wheel's rates $\dot{\theta}_r^i$,
- and steer angles θ_i^i .

These parameters are computed from a planar kinematic model based on ideal rolling assumptions and the features of the instantaneous center of rotation. This assumption is generally used during the definition of the low-level control of rover motion as it is difficult at this stage to take into account the local ground surface map.

Assuming that the rover has to follow a desired trajectory $(x_d(t), y_d(t))$ given on the horizontal plane and the rover x direction is tangent to this path, the forward velocity and the yaw rate can be written as follow

$$\begin{cases} v_{p,x}^d = \sqrt{\dot{x}_d^2 + \dot{y}_d^2} \\ \omega_{p,z}^d = \frac{\dot{x}_d \ddot{y}_d - \dot{y}_d \ddot{x}_d}{\dot{x}_d^2 + \dot{y}_d^2} \end{cases} \quad (45)$$

Assuming the rotation center along the y -axis and an ideal-rolling, the steer angle of the i th wheel is given by

$$\theta_i^i = \arctan \frac{x_c^i \omega_{p,z}^d}{v_{p,z}^d - y_c^i \omega_{p,z}^d} \quad (46)$$

and its rate by

$$\dot{\theta}_r^i = \sqrt{(v_{p,z}^d - y_c^i \omega_{p,z}^d)^2 + (x_c^i \omega_{p,z}^d)^2} \quad (47)$$

with (x_c^i, y_c^i) the wheel center coordinates in the platform frame \mathcal{P} .

D Contact parameters

k_x	0.01 s.mm ⁻¹
k_y	0.01 s.mm ⁻¹

Table 1. Kineto-Static contact model parameters

Static friction coefficient	0.8
Dynamic friction coefficient	0.6
Stiction transition velocity	80 mm.s ⁻¹
Friction transition velocity	160.0 mm.s ⁻¹

Table 2. Adams friction model parameters