Wheel torque control for a rough terrain rover

Pierre Lamon, Ambroise Krebs, Michel Lauria, Roland Siegwart
Autonomous System Lab
Swiss Federal Institute of Technology (EPFL)
Lausanne 1015, Switzerland
{pierre.lamon, ambroise.krebs, michel.lauria, roland.siegwart}@epfl.ch

Steven Shooter
Mechanical Engineering Dpt.
Bucknell University
Lewisburg, PA 17837, USA
shooter@bucknell.edu

Abstract - **Navigating in rough terrain is a complex task that requires the robot to be considered as a holistic system. Algorithms, which don't consider the physical dimensions and capabilities of the mobile robot lead to inefficient motion and suffer from a lack of robustness. A physical model of the robot is necessary for trajectory control. In this paper, quasi-static modeling of a six-wheeled robot with a passive suspension mechanism is presented together with a method for selecting the optimal torques considering the system constraints: maximal and minimal torques, positive normal forces. The aim of this method is to limit wheel slip and to improve climbing capabilities. The modeling and the optimization are applied to the Shrimp rover.**

I. INTRODUCTION AND MOTIVATION

A lot of trajectory planning and control algorithms don't consider the physical dimensions and capabilities of the mobile robot within its environment. In rough terrain, these algorithms lead to inefficient motion and suffer from a lack of robustness. Navigating in rough terrain is a complex task which requires the robot to be considered as a holistic system.

For wheeled rovers, the motion optimization is somewhat related to minimizing slip. Minimizing wheel slip not only limits odometric error but also reduces the overall energy consumption and increases the robot's climbing performance. A good distribution of wheel speeds and torques is necessary to fulfill this goal. A few publications concerning terrain estimation and physics-based motion planning in rough terrain can be found in the literature. A good overview of physics-based control and terrain modeling techniques is presented in [1,2,3,7]. Time optimal planning considering a dynamic model can be found in [4].

The speed of an autonomous rover must be limited in rough terrain in order to avoid high shocks on the structure and for safety reasons. Furthermore, the computational cost is usually quite high (image processing, path planning, obstacle avoidance, etc.) and the onboard processing power is limited. This requires the rover to move slowly. In this range of speeds (typically 5 to 20 cm/s), the dynamic contributions can be neglected and a quasi-static model is appropriate. Such a model can be solved for contact forces and motor torques knowing the state of the robot and the wheel-ground contact points.

The Autonomous System Lab developed a six-wheeled off-road rover called Shrimp which shows good climbing capabilities because of its passive structure. A detailed description can be found in [5]. This robot can adapt to a large range of obstacles and therefore can move smoothly across rough terrain. This behavior allows limited wheel slip. Nevertheless, further effort has to be done at the controller level. A physical quasi-static model of the robot must be developed in order to optimally control the torques on the wheels. Furthermore, a good knowledge of the wheel-ground interaction is required in order to set a slip probability for each wheel. This will facilitate probabilistic sensor fusion between 3D-odometric information [6] and other sensors such as inertial measurement unit, laser scanner and vision.

In this paper we will first describe the rolling resistance and slip models we are using. The quasi-static model of the Shrimp and a mobility analysis are presented in section 3 and 4. Section 5 will be dedicated to the method used for selecting the optimal torques considering the system constraints. The experimental results are presented in section 6 and section 7 will be dedicated to future work and conclusion.

II. WHEEL SLIP AND ROLLING RESISTANCE

The intent is to formulate a holistic model of a robot to optimize the control of the wheel motor torques to minimize wheel slip. Therefore it is helpful to review the governing equations on wheel slip and rolling resistance. These equations are later incorporated into a quasi-static model of a robot. Figure 1 shows the common forces acting on the wheel of a mobile robot.
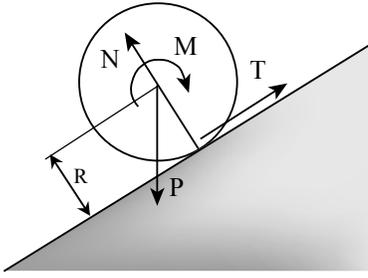
P : external wheel joint force   T : friction force
N : normal force   R : wheel's radius
μ_o : static friction coefficient   M : motor torque
μ : dynamic friction coefficient

Figure 1. Acting forces on a wheel

The wheel is balanced if the friction force fulfils the inequation:

$$F_{static} \leq \mu_0 \cdot N \qquad (1)$$

This case represents static friction.
If the static friction force can't balance the system, the wheel slips and the friction force becomes:

$$F_{dynamic} = \mu \cdot N \qquad (2)$$

In order to avoid wheel slip, the friction force which depends directly from the motor torque, M, should satisfy the equation (1)

$$T = \frac{M}{R} = F_f \leq \mu_o \cdot N \qquad (3)$$

The above equations suggest that there are two ways to reduce wheel slip. First, assume that μ_o is known and set:

$$T \leq \mu_o \cdot N \qquad (4)$$

In fact, it is difficult to know μ_o precisely because it depends on the kind of wheel-soil interaction. During exploration, the kind of soil interacting with the wheels isn't known which makes μ_o impossible to pre-determine. Anti-lock break systems in automobiles sense slip and then compensate T until slip is not sensed anymore. But in this case slip has already occurred.

Another way to avoid wheel slip is to first assume that the wheel does not slip. It is then possible to calculate the forces T and N as a function of the torque and the result is optimized in order to minimize the ratio T/N. Accounting for the previous assumption:

$$\frac{T}{N} = \frac{\mu_n \cdot N}{N} = \mu_n \qquad (5)$$

μ_n is similar to a friction coefficient. In minimizing this ratio, then minimizing μ_n, we optimize our chances that this coefficient is smaller than the real friction coefficient μ_o. If this is the truth, there is no slip. Therefore, it is possible to minimize the ratio T/N without knowing the real static friction coefficient. The second method is used here, because it is more robust. However, determination of parameters T and N require a model of the mobile robot.

Rolling resistance is another important aspect to the quasi-static model, and is therefore reviewed here. A static model balances the forces and moments on a system to remain at rest or maintain a constant speed. Such a system is an ideal case and does not include resistance to movement. The rolling resistance is introduced in order to complete the model. This results in a quasi-static model. For a motorized wheel, we have:
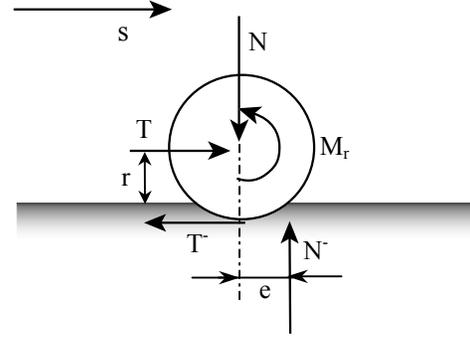
T, N : tangential/normal force   s : movement of the wheel
T⁻, N⁻ : ground reaction forces   e/r : rolling resistance parameter
R : wheel's radius M_r : friction torque

Figure 2. Rolling resistance on a motorized wheel

The friction torque, or rolling resistance torque, is opposed to the movement (Hertz-Föppl model [8]):

$$M_r = R \cdot T = 0.15 \sqrt{\frac{R}{l \cdot E}} N^{3/2} \qquad (6)$$

where l is the length of the rectangular contact patch and E is the reduced elasticity module described by :

$$E = \frac{2 \cdot E_1 \cdot E_2}{E_1 + E_2} \qquad (7)$$

where E_1 and E_2 are the elasticity of the wheel and the ground. This representation expresses the friction torque as dependant on the normal force applied on the wheel. A greater normal force results in a greater resistance torque which follows intuition.

Equation 6 is not linear which can be difficult for analytical solutions. It is therefore simplified as:

$$M_r = \alpha \cdot 0.15 \sqrt{\frac{R}{l \cdot E}} N \qquad (8)$$

where α is a coefficient for reducing the simplification error. α is identified after an iterative process that estimates the simplification error.

III. INFLUENCE OF THE DEGREES OF FREEDOM

The mobility of a rolling robot in straight motion should ideally be one, indicating that the robot can move in a constrained direction. Grubler's Mobility Equation in three dimensions can be described as:

$$MO = 6 \cdot n - 5 \cdot f_1 - 4 \cdot f_2 - 3 \cdot f_3 - 2 \cdot f_4 - f_5 \qquad (9)$$

where n is the number of mecanical parts and f_j the number of joints of each type (j=1,..,5, for example f_1: the number of pin joints, f_3: the number of spherical joints). The mobility equation is a guideline for determining if a system is statically determinate. Many real systems contain redundancy in links and joints resulting in hyperstatism. A four-legged table, for example, is statically indeterminate. More sophisticated modelling methods are required to analyze the distribution of forces in a hyperstatic system. Another approach is to model selective joints with additional degrees of freedom. Intelligent selection of these joints can minimize the error associated with a quasi-static solution. While the modeled kinematic chain is a simplification, it can be good enough to support motor control.

Example for the Shrimp

The mobile robot Shrimp has six powered wheels as shown in Figure 3. One wheel is rigidly attached to the body. In the front, a second wheel is effectively in the same plane with its position governed by a linkage and a spring. On either side two-wheeled bogies provide lateral stability. The position of the parallel-linkage bogie is passively articulated from reaction of the terrain.
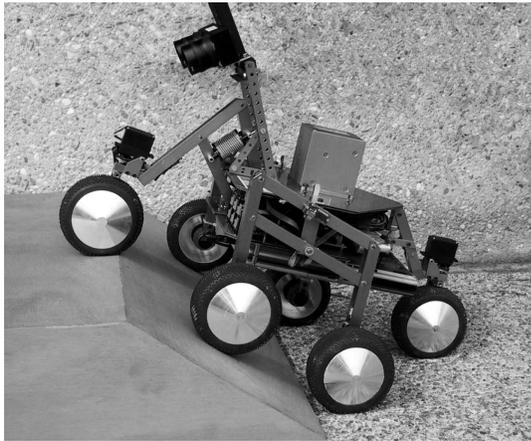

Figure 3. The Shrimp structure.

In a first step, we can consider the wheel-ground contacts as spherical joints and all the pin joints in the mechanism as 1DOF revolute. For the Shrimp, the calculation of the mobility using equation (9) is –20 rather than 1. The system is, therefore, significantly hyperstatic and requires a modified model for a possible quasi-static solution. Two significant modifications to joint degrees of freedom assists the model.

The first one involves the representation of the wheel-ground joint mobility. For a standard wheel without slip, the joint that represents the wheel-ground contact can be modelled as a spherical joint allowing three degrees of freedom (rotations about the three axes). Motor torque on the wheels will directly affect the forces in that contact plane. Lateral forces are not influenced by the motor torque. Therefore, the system was modelled with the lateral forces being carried by the wheel fixed to the body and the wheel on the front fork. The wheels on the bogies were modelled with no resistance in the lateral direction (4 degrees of freedom).

The second modification acts on the representation of the redundant kinematic chains. It is possible to model selected joints on redundant kinematic chains with greater degrees of freedom. This results in force being transmitted through direct flow patterns. Because the model is being used to optimise motor torques, inaccuracies in the internal linkage forces can have minimal effect.

Figure 4 shows the resulting kinematic model of the Shrimp mobile robot. The numbers at the link connections indicate the degrees-of-freedom of that joint.
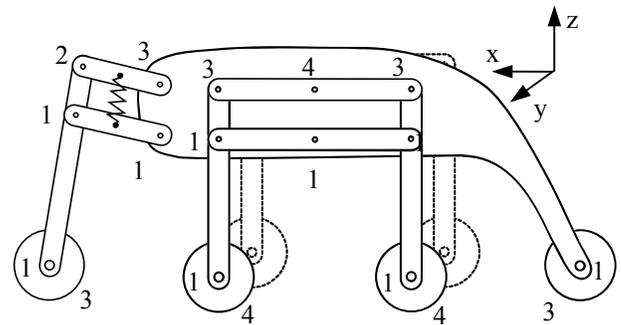

Figure 4. Final representation of the mobility of the joints.

The final mobility can then be calculated using equation (9) to produce:

$$MO = 6 \cdot 18 - 5 \cdot 14 - 4 \cdot 1 - 3 \cdot 7 - 2 \cdot 6 = 1 \qquad (10)$$

IV. QUASI-STATIC MODEL OF A ROBOTIC SYSTEM

For a 3D static model, 6 equations (3 torques and 3 forces) are applied to each body, containing ground reaction forces, gravity forces (weight) and external forces. The resulting model that is used for motion control also includes consideration of wheel resistance and wheel slip. Dynamic

forces are considered to be negligible because the speed is low. Therefore, the model is referred to as quasi-static.

For the model we assume that the wheel-ground contact angles are known. They can be computed while the robot is moving using the technics presented in [1] and [6]. An alternative method using tactile wheels can be found in [9]. It has the advantage to provide the contact points for static conditions also.

Model of the Shrimp

The Shrimp has 18 parts and is characterised by 6·18 = 108 independent equations describing the static equilibrium of each part and involving 14 external ground forces, 6 internal wheel torques and 93 internal forces and torques for a total of 113 unknowns. The weight of the fork and the bogies link has been neglected whereas the weight of the main body and the wheels is considered.

Of course, it is possible to reduce this set of independent equations because we have no interest in implicitly calculating the internal forces of the system. The variables of interest are the 3 ground contact forces on the front and the back wheel, the 2 ground contact forces on each wheel of the bogies and the 6 wheel torques. This makes 20 unknowns of interest and the system could be reduced to 20 - (113 - 108) = 15 equations. This leads to the following matrix equation:

$$M_{15x20} \cdot \vec{U}_{20x1} = \vec{R}_{15x1} \qquad (11)$$

where M is the model matrix depending on the geometric parameters and the state of the robot, **U** a vector containing the unknown and **R** a constant vector.

<center>V. OPTIMIZATION</center>

The controlable inputs of the system are the six wheel torques. Since there are five more unknowns than equations it is possible to write an equation expressing the torques as linearly dependant. The 14 other equations define the external forces as a function of the torques.

The model of the Shrimp is indeterminate because there are less equations than variables and the set of solutions is of dimension five (number of wheels -1). The goal of the optimization is to minimize slip. This can be achieved by maximizing the traction forces, which is equivalent to minimizing the function $\max(T_i / N_i)$ for the wheels [1].

Since it is difficult to do reasoning in five dimensions, a simpler robot referred to as ThreeWheels (see Figure 5) is used for presenting our optimization algorithm. This process will then be extrapolated for the complete model.
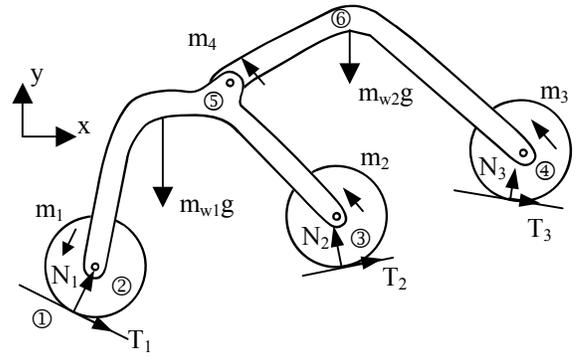

Figure 5. The ThreeWheels 2D model. This rover belongs to the passively suspended robots family. m_4 is a non-controllable torque generated by a torsion spring with known characteristics.

The model of the ThreeWheels rover has nine unknowns: two forces and one torque on each wheel (m_4 is known and directly depends on the geometry) and seven equations: three global equations, one torque equation for each wheel and one torque equation for the fork. That means that the solutions space is of dimension two and that m_1, m_2 and m_3 are linearly dependant. Equations (12) express the forces on the wheels and equation (13) the torque of the first wheel as functions of m_2 and m_3.
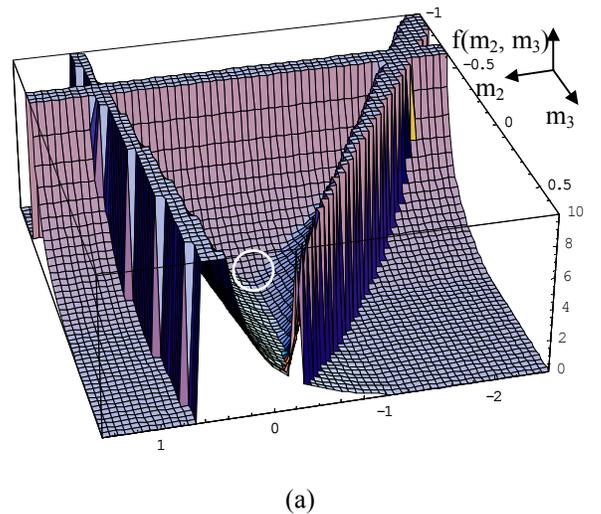
$$N_i = \alpha_{1i} \cdot m_2 + \beta_{1i} \cdot m_3 + \gamma_{1i} \qquad (12)$$

$$T_i = \alpha_{2i} \cdot m_2 + \beta_{2i} \cdot m_3 + \gamma_{2i} \quad \text{with } i = 1,2,3$$

$$m_1 = \varepsilon_1 \cdot m_2 + \varepsilon_2 \cdot m_3 + \delta \qquad (13)$$

$$f(m_2, m_3) = \max\left(\frac{T_1}{N_1}, \frac{T_2}{N_2}, \frac{T_3}{N_3}\right) \qquad (14)$$

The solution space of the ThreeWheels rover is depicted in Figure 6a. It corresponds to the function f defined by the equation (14), which is the function to minimize.
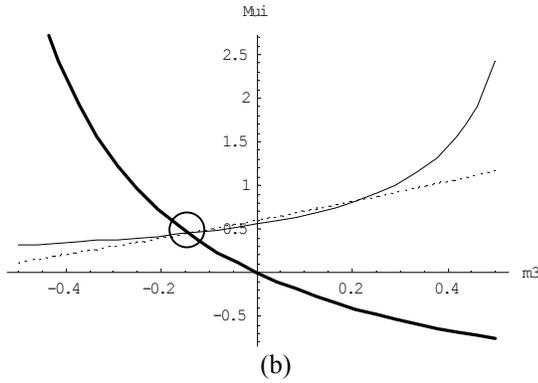

(a)

(b)

Figure 6. Solution space for the ThreeWheels rover (see Eq. 14): max(Mu$_1$, Mu$_2$, Mu$_3$). The functions Mu$_1$, Mu$_2$, Mu$_3$ are hyperbolic and a linear optimisation process is not possible. (a) Optimal solution (circled) minimizing slip and fulfilling the N$_i$ > 0 constraint. (b) Cross section of figure (a) for m$_2$ = -0.3061. One can see that the optimal solution (circled) corresponds to equal Mu's.

Since the system of equations is non linear, a numerical method is implemented. Our optimization method uses a combination of different algorithms and is depicted in Fig 7. Firstly the Equal Torques solution is checked versus the following constraints:

a. Motors saturation : the torques of the optimal solution must be smaller than the maximal possible torque.
b. Normal forces : the normal forces N$_i$ must be greater than zero. The asymptotes of the hyperbolic functions in Fig 6a define the sign inversion limit.

If this solution is valid, it is taken as the initial solution for the Fixed Point optimization (A). If it doesn't fulfill the constraints, a valid initial solution is computed using the Simplex Method (B). The optimal solution is then provided either by (A) or the Gradient optimization (C).
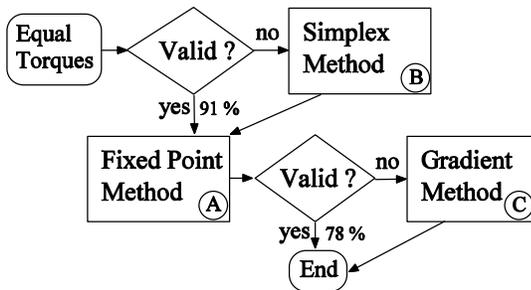

Figure 7. Optimization algorithm.

Now the individual modules are explained in more detail.

A. Fixed Point optimization

This optimization method is based on the fixed point algorithm. The aim of this algorithm is to numerically find an intersection of curves when an analytical solution is hard to obtain. This is the case when searching for solutions with equal Mu's. The corresponding flow chart along with its explanation is presented in Figure 8.
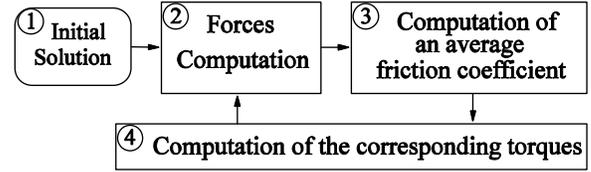

Figure. 8 : Fixed point based algorithm. The quasi-static model (2) is solved with an initial set of torques (1). Block (3) computes an average friction coefficient based on the computed forces (output of module 2). The corresponding torques are computed (4) and fed again in module (2). Twenty iterations are sufficient for convergence.

This algorithm is not computationally expensive and provides good results for most cases. Nevertheless, it diverges sometimes and doesn't account for the before-mentioned constraints. This can lead to torques that cannot be provided by the motors.

B. Simplex method

This method is based on the Simplex algorithm which solves linear programs in a constrained solution space. The Simplex method tries to maximize an object function considering a set of constraints on the variables. We choose the object function *g* defined in equation (15) as the object function. This function has been chosen because it tends to minimize the ratio T/N.

$$g = \max(\sum_i N_i) \qquad (15)$$

Furthermore, the function *g* is linear because it is a linear combination of the torques. The solution provided by this method fulfils the constraints and will be used as a starting point for the Incremental optimisation.

C. Gradient optimization

This algorithm seeks for an optimum in the constrained solutions space given a known valid initial solution. The gradient optimization is similar to the potential field method: at each step the gradient is computed and the next solution is generated following the maximum slope.

VI. EXPERIMENTAL RESULTS

The optimization for the 3 dimensional Shrimp is similar to the method presented in the previous section. The solution space has now five dimensions and one has to account for 18 constraints (with i = 1 … 6).

$$m_i < MaxTrq \qquad m_i > -MaxTrq \qquad N_i > 0 \qquad (16)$$

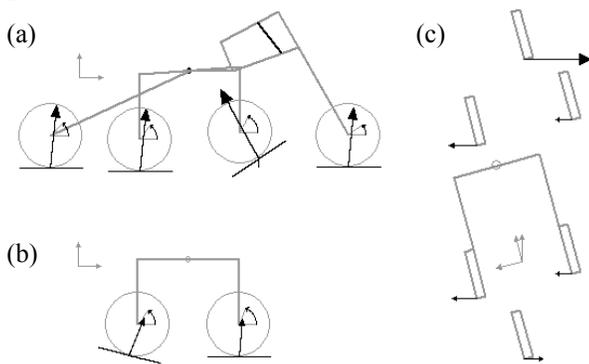An example of computed forces and torques is depicted in figure 9.

Figure 9. Forces and torques computed by the optimization procedure. The forces are expressed in the global frame of reference. The user of the interface can change the state of the robot and the contact angles of the wheels interactively. (a) Side view: the pitch, the front fork and the left bogie angles can be modified (b) Right bogie view: the angle of the right bogie can be modified on this view. (c) Decomposed view from rear: the roll angle can be modified. In this view, the arrows represent the projections of the reaction forces in the global frame of reference.

The optimization algorithm has been tested for around 20 thousand states generated automatically considering different angles for each input parameter. **80%** of the computed torques correspond to a friction coefficient smaller than 0.6 (a tyre on a dry road has a coefficient of 1). This will certainly allow to limit the slip of the wheels in most of the cases.

The execution times for the algorithms A, B and C are 6 ms, 5 ms and 20 ms respectively (1.5 GHz processor). The worst case is about 31 ms. However, as it can be seen in Fig. 7, the majority of the states are handled in 6 ms.

VII. FUTURE WORK AND CONCLUSION

In this paper a quasi-static model of a six-wheeled rover together with an optimization method which minimize slip have been presented. The simulations show promising results and the system is ready to be implemented on the rover for real testing. The needed processing power is relatively limited and online computation should be easily accomplished.

The next step will be to use the computed forces and torques in order to set a slip probability for each wheel: the less pressure on the wheel, the more likely the wheel slips. This is a valuable information for probabilistic multi-sensors fusion. The knowledge of wheel-ground interaction will also give a clue on how to correct for wheel diameter changes due to tire compression.

REFERENCES

[1] Iagnemma, K., Shibly, H., Rzepniewski, A., and Dubowsky, S., "Planning and Control Algorithms for Enhanced Rough-Terrain Rover Mobility," Proceedings of the Sixth International Symposium on Artificial Intelligence, Robotics and Automation in Space, i-SAIRAS, 2001.

[2] Iagnemma, K., Shibley, H., Dubowsky, S., "On-Line Terrain Parameter Estimation for Planetary Rovers," Proceedings of 2002 IEEE International Conference on Robotics and Automation (ICRA 2002), Washington D.C, May 11- May15, 2002.

[3] Iagnemma, K., and Dubowsky, S., "Mobile robot rough-terrain control (RTC) For planetary exploration", Proceedings ASME Design Engineering Technical Conferences, Baltimore, Maryland, September 2000.

[4] Shiller, Z., Gwo, Y.-R., "Dynamic motion planning of autonomous vehicles", IEEE Transactions on Robotics and Automation, Vol. 7, Issue: 2, April 1991

[5] Siegwart R., Lamon P., Estier T., Lauria M., Piguet R., "Innovative design for wheeled locomotion in rough terrain", *Journal of Robotics and Autonomous Systems*, Elsevier, vol 40/2-3 p151-162.

[6] Lamon P., Siegwart R., "3D-Odometry for rough terrain – Towards real 3D navigation", Proceedings of IEEE International Conference on Robotics and Automation (ICRA 2003), Taipei, May 2003.

[7] Hung, M.-H., Orin, D. E., Waldron, K. J., "Efficient Formulation of the Force Distribution Equations for General Tree-Structured Robotic Mechanisms with a Mobile Base", IEEE Transactions on Systems, man and Cybernetics, Part B: Cybernetics, Vol 30, No 4, August 2000.

[8] Kalker J.J., "Three dimensional elastic bodies in rolling contact", Kluwer Academic Publishers, Dordrecht, 1990.

[9] Lauria, M., Piguet, Y. and Siegwart, R., "Octopus - An Autonomous Wheeled Climbing Robot", In Proceedings of the Fifth International Conference on Climbing and Walking Robots, Published by Professional Engineering Publishing Limited, Bury St Edmunds and London, UK.