DINAME 2013 - Proceedings of the XV International Symposium on Dynamic Problems of Mechanics
M.A. Savi (Editor), ABCM, Buzios, RJ, Brazil, February 17-22, 2013

Planetary Rover Optimization Tool (ROT)

Alexandre Carvalho Leite[1], **Bernd Schäfer**[1]

[1] Deutsches Zentrum für Luft- und Raumfahrt, Institute of Robotics and Mechatronics, Weßling, 82205, Germany (e-mail: alexandre.carvalholeite@dlr.de, bernd.schaefer@dlr.de)

Abstract: A planetary rover optimization tool has been developed at DLR since 2009, where a lot of modeling, simulation and optimization effort were injected into this initiative to computer aided rover design. This paper shows the main characteristics of this tool concerning optimization of an ExoMars-type suspension rover in multiple scenarios using evolutionary algorithms, distributed computation, and enhanced wheel-terrain contact models embedded in a multibody simulation model. The tool was already extensively tested and employs meaningful objective functions specially designed to capture the dynamic performance of wheeled rovers in extreme environments.

Keywords: Scenario-oriented optimization, rover modeling, rover simulation, multi-body simulation, contact modeling

INTRODUCTION

Wheeled vehicles are an important means to increase scientific return while roving on planetary surfaces. NASA's four Mars rovers (Sojourner, Spirit, Opportunity and the recently landed Curiosity) and the planned European ExoMars rovers (in 2018) are examples and challenging contributions to explore larger surface areas. However, future rover designs should efficiently rely not only on best engineering intuitions but also on powerful optimization tools. This contribution is expected to increase both, rover performance and exploration area. The design of planetary exploration rovers is an involved process including several conflicting system requirements. Mobility and locomotion, operations, environment, power, thermal, structures and mechanical environment characteristics are interrelated through a locomotion systems concept. This concept includes a suspension type and its geometric parameters. The former is defined by a human designer aided by some trade-off metrics and is a rather intuitive process. On the other hand, the synthesis of the geometric parameters considers dynamic/static analysis which can be done in a computer aided manner. Thus, we assume the following premise to automated synthesis of planetary rovers: a given suspension concept can be further improved by changing algorithmically its geometric parameters. This is done here in a scenario-oriented optimization tool specifically developed for planetary rovers; it relies basically on well tested numeric optimization and multi-body simulation tools.

Scenario-oriented optimization

Several characteristics of wheeled robots are directly affected by the terrain where it is driving. For example, ground clearance, static stability margin, and wheel width cannot be defined without considering respectively: the height of the obstacles, highest slope value on a realistic relief and the mechanical characteristics of the sand. These conditions are considered as parts of a scenario to be simulated in interaction with a rover. For simulation purposes, we compose the scenarios with obstacles (represented with triangle soups), a relief (digital elevation model), and terrain type (rigid as smoothed Coulomb friction, or deformable according with Bekker-Wong theory in Bekker (1956) and Wong (2001)). In the optimization point of view, a scenario is a constraint which is very meaningful but difficult to set up. The complexities of real scenarios have to be captured keeping the compromise among imposed computational overhead, representativeness, and to provide a consistent evaluation of different vehicles configurations. The shape of a real scenario could be accurately reproduced in a simulation environment with high resolution digital representations of the reliefs and obstacles, but the computational costs regarding collision detection, soil deformation, and contact forces computation would be impracticable even for a single simulation. Obstacles placed like in a real scenario could impose qualitative incomparable behaviors to different geometric configurations of a same suspension concept. It has to be guaranteed that a simulation scenario represents suitably a real scenario but is still able to: 1) generate simulation models with acceptable simulation times; 2) affect the dynamic behavior of the vehicles suspension in the same way for different geometric configurations. The first condition is stated because of a simple reason, a single simulation is just an iteration of an optimization process which can take more than 200 iterations. The second condition is inherent to the non-linearity of the obtained simulation model, the wheels of two different geometric configurations have to face the same obstacles and the same tracks in order to be compared with each other. If the last condition is not achieved, the optimization results will not be meaningful since the objective functions will return result for conditions which are in fact different.

We verified the scenario-oriented optimization by defining two scenarios (rigid and smooth terrain) where the rover drives straight ahead facing obstacles and a slope in the case of the rigid terrain. The simulation scenarios were composed in such a way that each rover of the optimization iterations would interact with the obstacle in the same locations and the following wheels (in the case of soft terrain) face the tracks left by the other wheels. The objective functions were overall mass, consumed power, accumulated sinkage, and dynamic stability as described in Schäfer and Leite (2011). The result in Fig. 1 shows that this is exactly as expected: a rover with reduced vehicle width and increased length to account for longitudinal stability. The width of the wheels were decreased and radius increased, these characteristics reduce the accumulated sinkage measure as they reduce the dimension of the tracks and increase the traveled path.

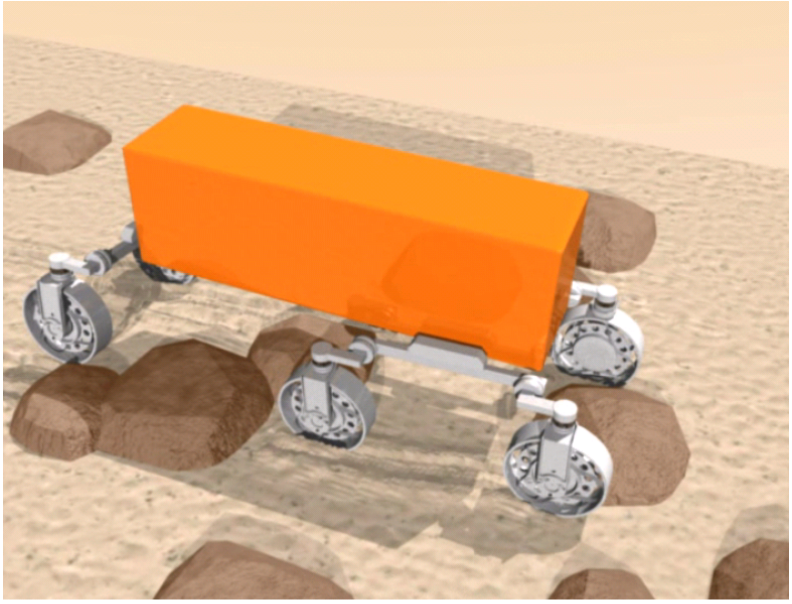

Figure 1 – Geometric configuration of an ExoMars-type suspension synthesized for verification of the scenario-oriented approach

Other scenarios were defined as presented in Fig. 2 in order to obtain a geometric configuration of the ExoMars-type rover which agrees with realistic planetary rover requirements. In this work we used six different objective functions evaluated in the seven different scenarios to increase robustness of the solution, each objective function is related to the respective scenario according to Tab. 1.

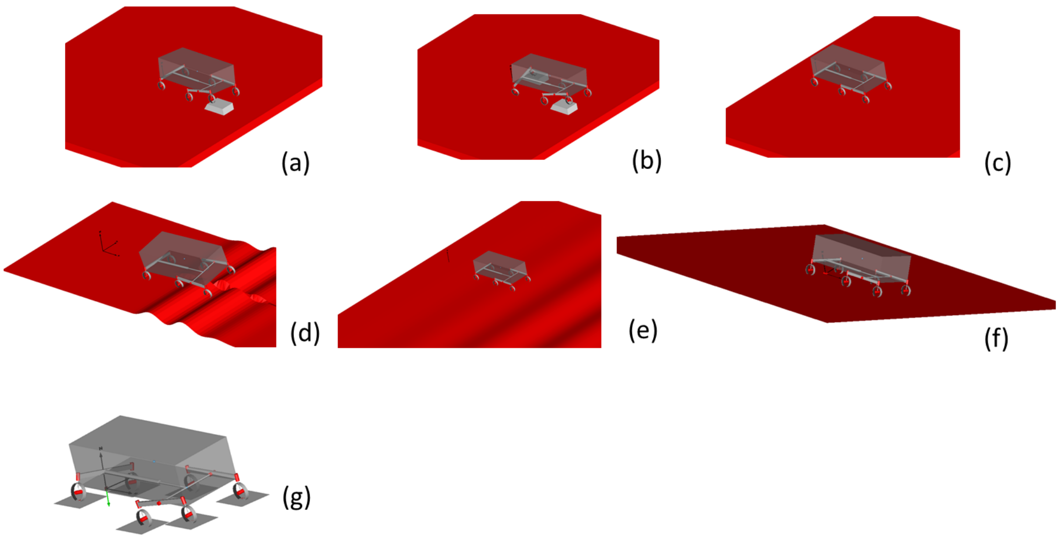

Figure 2 – Seven simulation scenarios selected: (a) one obstacle in soft soil; (b) two obstacles in soft soil; (c) no obstacle in soft soil; (d) staggered rigid terrain; (e) undulating rigid terrain; (f) rotating rigid plane; (g) elevating rigid terrain

Even obeying the compromises stated before to choose suitable scenarios, computational cost is still high. The optimization procedure was implemented with distributed computation in our cluster at DLR, it reduced the overall computational cost to that of the most expensive simulation alone. In a normal implementation it would take approximately an optimization time of $n_i \times \sum_{X \in \{A,\ldots,G\}} t_X$ where n_i is the number of iterations of the optimization and t_X is the simulation time of a scenario indexed by X. With distributed computation the time reduces to approximately $n_i \times \max_{X \in \{A,\ldots,G\}} t_X$, which saves several days of optimization time in a 3GHz Intel Xeon CPU.

As the main result of this work we present the Rover Optimization Tool (ROT), see Fig. 3, including complex terrain composition, contact modeling, multi-scenario optimization setup, and post processing analysis functions. It is composed of Matlab APIs, Dymola/Modelica models, and a Matlab GUI (Graphical User Interface) to integrate the composed simulation setup with the DLRs optimization tool, MOPS (Multi-Objective Parameter Synthesis).

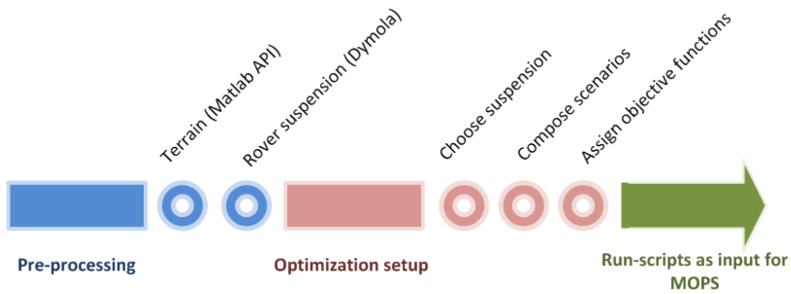

Figure 3 – Schematic of rover design process using ROT

Table 1 – Objective functions assigned to each of the 7 scenarios

Scenario	Overall mass	Average power	Accumulated sinkage	Stability	Traveled distance	Attitude path
A	✓	✓	✓	-	-	-
B	-	✓	✓	-	-	-
C	-	✓	✓	-	-	-
D	-	-	-	✓	✓	-
E	-	-	-	✓	✓	-
F	-	-	-	✓	-	-
G	-	-	-	✓	-	✓

Characteristics of the ROT

The ROT was conceived to be used either in the concept or engineering development phase of a planetary rover. In the concept development phase different suspensions can be simulated and compared to select the suitable candidates. These candidates will be further explored (individually optimized) and define the kinematic concept. A definitive kinematic concept is chosen when two or more optimized candidates are compared and a trade-off is performed. During engineering development the whole set of specifications is available and can be used as additional inputs to increase the number of scenarios and refine the parameters of a given kinematic concept.

ROT is a toolset which covers modeling, simulation and optimization functionalities. The resources are implemented in Modelica and MOPS respectively under the commercial tools Dymola and Matlab, they were developed specifically for this toolset and are described as follows:

RCM stands for Rough Terrain Contact Models. It implements the geometry of the contact models for soft and rigid terrain using the low-level functions already implemented in SODY and TMec.

SODY stands for SOLID-Dymola interface. SOLID is a collision detection library mainly developed for video game applications and is used here to detect the witness points between each wheel and terrain components (relief and rocks). This interface includes smoothing techniques specifically implemented to make Dymola simulations more stable when interfacing with SOLID.

TMec stands for Terramechanics library, (Leite and Schäfer, 2010a). All force computations are implemented in this library using Bekker-Wong theory extended for three dimensional wheel-soil interaction.

ROT GUI is the graphical user interface to create Dymola/Modelica simulation models from Matlab. It generates Modelica code which can be opened in Dymola, verified, and used for either optimization or general simulation purposes.

ROT API is a collection of functions used mainly in MOPS run-scripts and enables the generation of CAD models of reliefs and stones.

ROT Visualization is a collection of Python scripts which imports Dymola simulation results into Blender 3D to render animations with increased quality.

Figure 4 shows how an optimization setup is generated. In this illustration, a vehicle model is given as input for ROT. The first step is to open this model in ROT and define which parameters can be tuned during optimization as well as their ranges and nominal values. The second step consists in creating a new scenario, assign objective functions to be evaluated in the context of that scenario, insert and place obstacles, define and configure terrain. This step can be repeated as long as additional scenarios are needed, in Fig. 4 just one scenario is created. The third step is the generation of simulation models for each terrain associated to the vehicle and optimization setup which can be called independently in MOPS.

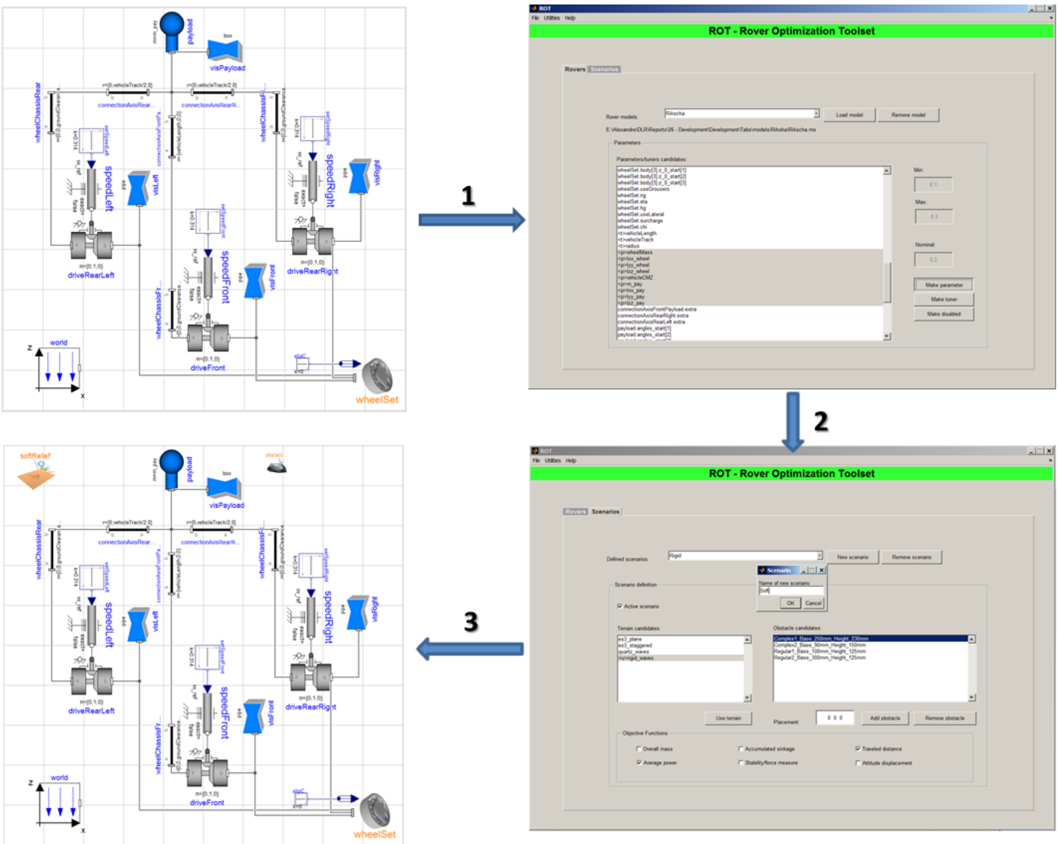

Figure 4 – Optimization setup in ROT in three basic steps

Some of the ROT's API functions can only be called inside a MOPS run-script workspace, because they deal with the data structure provided by Dymola simulations started by MOPS. The MOPS-specific ROT API functions evaluate objective functions. Terrain components (relief, obstacles, and soil types) can be created by another three API functions, they are normally called in the Matlab's base workspace to include new elements to the folders in the respective ROT GUI's subdirectories. In other words, they include new components to the terrain components database. These terrain components can be later assigned to some scenario.

Application of the ROT

The three-step procedure described in the previous chapter was employed to the optimization of two different wheeled rovers, ExoMars and Rocker-Bogie suspension types. The design parameters are geometric dimensions of wheels, bogies, contact area of the vehicle. Each mechanical part is described as a simple primitive (e.g. hollow cylinder for wheels, rectangular prism for bogies) and as such can simplify too much the mass distribution. To avoid inconsistencies in comparison with reality, additional masses are inserted to account for certain mass limits imposed by cabling, actuators, sensors and minor mechanical details, see Tab. 2.

In ROT more than one objective function can be assigned to some scenario. Multiple scenarios are common in order to setup a meaningful rover optimization problem, which gives several objective functions to be evaluated and optimized.

Table 2 – Additional masses

Description	Additional mass [kg]	Quantity [-]	Location [-]
Drive joint	2.60	6	wheels
Front bogie cables and electronics	2.61	2	front bogies
Transversal bogie cables and electronics	1.82	1	transversal connection
Deployment and steering joints	3.12	6	deployment/steering joints

These objective functions can be aggregated in many ways (e.g. sum of their normalized squared values). Although, the convergence of the aggregated objective functions in a large space of design parameters does not give reasonable insight about the convergence. Figures of merit can be used to understand the coupling among several objective functions and design parameters, but each evaluation pair is chosen carefully to avoid a large number of figures of merit in the analysis process. As an example, Fig. 5 shows two figures of merit from the optimization of the ExoMars Rover: Fig. 5-left confirms the notion of sinkage increasing as dependent on vehicle mass, but the trend is not well behaved because of the influences of other parameters in the weight distribution; Fig. 5-right shows an optimal point which is a true compromise: high values of mass are not desirable, small values of mass are desirable but enforce instability in some direction, height of payload centre of mass should be as high as possible as long as stability is preserved. All these factors contribute to the convergence patterns in Fig. 5 and are used for verification in case the optimization progress seems to be meaningful.

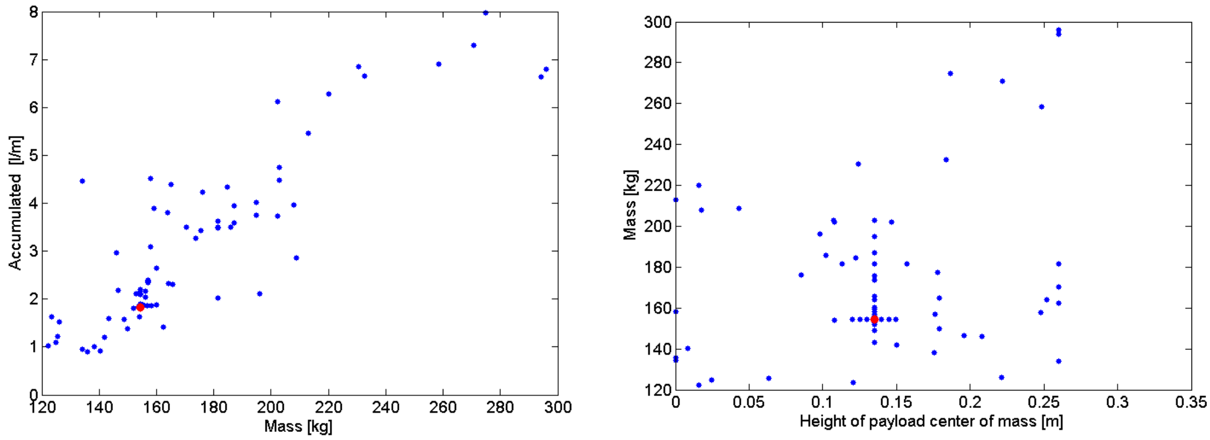

Figure 5 – Convergence relations: Objective functions (left) and objective function versus optimizing parameter (right)

The same scenarios and objective functions were employed to optimize ExoMars and Rocker-Bogie suspension concepts. The output provided by ROT (ROT Visualization) shows the final solutions for each suspension in Fig. 6. In this case, the ExoMars concept was capable to achieve the best values in the optimization functions and after some trade-off can be chosen as the vehicle to be further optimized. A comparison of design parameters and their optimal values can be seen in Tab. 3.

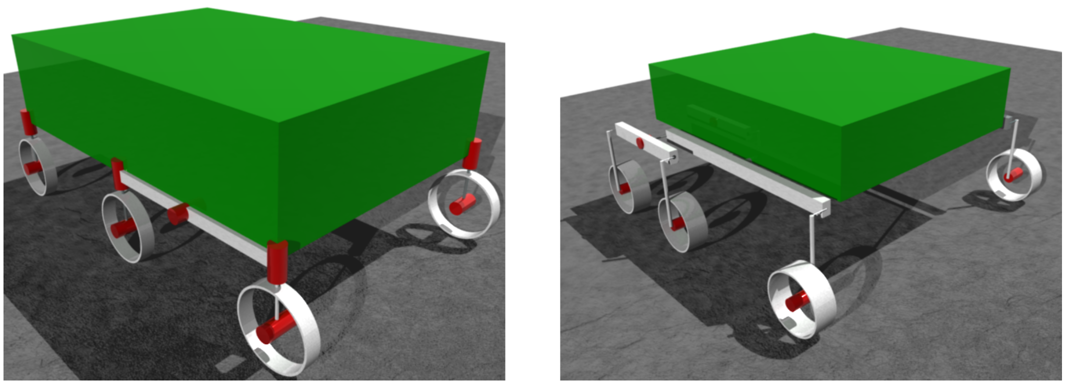

Figure 6 – Optimized results: ExoMars (left) and Rocker-Bogie (right)

Table 3 – Optimized ExoMars dimensions

	Nominal [m]	Optimized [m]
Distance from rear wheels to center of the front bogies	1.08	1.071
Length of front bogies	0.64	0.715
Wheel width	0.1	0.05
Wheel radius	0.125	0.125
Distance from wheel axis to bogie	0.205	0.198
Vehicle track	1.0	0.958
Distance from suspension to CM of the box	0.3	0.174

Small changes can be noted in some design parameters, they are due to constraints which confine the possible solutions in a reasonably small range. However, useful configurations could still be found in this reduced domain, which means sensitivity of the objective functions with respect to the terrain characteristics.

Lessons learned by applying ROT to the ExoMars type rover

System requirements of a real rover restrict the solution space in such a way that just small changes around the nominal one are meaningful. The volume of the rover before deployment limits the width and length of the vehicle. The height is also limited but it is defined not just by the design of the locomotion part (suspension and wheels). Load shocks are frequently considered to design wheels and bogies, bending moments are computed having these load shocks as reference values. Ground clearance is the minimal distance from ground to suspension, this distance is chosen to avoid collisions with obstacles which may cause damage to the instruments placed over the suspension. Terrain slopes and dimensions of crevasses specified for the traveled terrain also reduce the range of acceptable values for wheel and vehicle dimensions.

Note that the contact area of the vehicle cannot exceed some value because of the specified volume before deployment. But it cannot assume small values because static stability will no longer be possible on steep terrain. These situations define the possible range of variation of the design parameters. Bending deformations restrict the smallest dimensions of the cross-sectional area of the bogies (Leite and Schäfer, 2011). It constraints this property in such a way that allowable changes are negligible in the cross-sectional area of a rectangular beam. In the case of the ExoMars Rover, the mechanical parts of the suspension were modeled as filled rectangular aluminum beams. In this case there is not so much improvement during optimization and topological/material optimization of the mechanical connections of the suspension could reduce significantly the mass of the rover. The specified ground clearance is crucial to define the radius of the wheels, just because this parameter is a function of the wheel diameter and height of the steering/deployment mechanism. Since the design of this mechanism depends on several factors not taken into account during the optimization, it practically makes the wheel radius proportional to the ground clearance value.

Mass, power and stability are objective functions which are always present in our optimization setups (Leite and Schäfer, 2010b). A small number of design parameters makes the problem easier to understand and improves the optimization convergence. Scenarios should possess simple properties and clearly evaluate specific objective functions. A nominal solution is useful to be used as reference to the normalization of objective functions, this is very important for convergence. It is possible improvement and worsening of the same objective function when evaluated in two different scenarios, it is although interesting to keep such objective functions to cover a high number of performance indices. Some scenarios evaluate quite well some objective functions, a rotating plane for example can just evaluate the static stability objective functions. It is highly desirable to have scenarios where more than one objective function can be evaluated, it increases the robustness of the result. Power consumption for example is not the same for rigid terrain and soft soil with flat relief pattern. If one changes just the width of wheels, the results for power consumption would be almost the same in rigid terrain, but not in soft soil. Changes in mass and wheel dimensions in soft soil impact directly sinkage, reaction torque and available soil traction. In other words, the same objective functions has different sensitivities with respect to the chosen scenario.

The optimization process needs inputs from requirements' documents, but it also provides important inputs to design other subsystems. The static stability for example, can only be achieved if the whole assembly carried by the suspension has a centre of mass position as proposed by the optimized solution. Batteries and solar panels can also be designed taking into account the optimized power consumption values.

Concluding comments

In this paper the Rover Optimization Tool - ROT has been presented. Some of its functions were described as well as setups under the structure of the tool. Outputs of the ROT were shown as figures of merit and visualization output. Modeling, simulation, optimization setup, automatic generation of simulation models for scenario-oriented optimization, distributed computation, visualization of animations, and log data from the optimization process are included in the effort to develop ROT. The tool is currently in use in DLR for multiple purposes and in continuous development to increase fidelity, robustness and speed of simulations, and to make the rover optimization process faster through additional features in distributed computation. Multibody simulations of vehicles need in general tuning of several parameters to guarantee meaningful and robust simulations in complex terrains. The robustness of the simulations is due to constant development of RCM, SODY, and TMEC. Initially, an optimization run with two or three simple scenarios could take five days in a 3GHz Intel Xeon CPU. After specific implementation for computation in a cluster (each of the 32 nodes has a 3GHz double-core Intel Xeon CPU), several complex scenarios can be included in an optimization setup and provide results (optimization logs, figures of merit, and animations) in about four days.

ACKNOWLEDGMENTS

Parts of the work on contact modeling development were performed in conjunction with the HGF alliance project Planetary Evolution and Life (HGF - Helmholtz-Gemeinschaft der Großforschungsanstalten). And parts of the work leading to the parameter estimation results have received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under Grant Agreement n° 262744. The authors wish to gratefully acknowledge the support by HGF and EC.

REFERENCES

Bekker, M. G., 1956, "Theory of land locomotion", Ed. 1, The University of Michigan press, Ann Arbor.

Leite, A. C. and Schäfer, B., 2011, "On multi-objective optimization of planetary exploration rovers applied to ExoMars-type rovers", Proceedings of the ASTRA 2011, ESA/ESTEC. 11th Symposium on Advanced Space Technologies in Robotics and Automation, 14. April 2011, Noordwijk, The Netherlands.

Leite, A. C. and Schäfer, B., 2010a, "A comprehensive wheel-terrain contact model for planetary exploration rover design optimization", Proceedings of the Joint 9th Asia-Pacific ISTVS Conference and Annual Meeting of Japanese Society for Terramechanics, 27. Sept. - 30. Sept. 2010, Sapporo, Japan.

Leite, A. C. and Schäfer, B., 2010b, "Mass, Power and Static Stability Optimization of a 4-Wheeled Planetary Exploration Rover", Proceedings of the EngOpt 2010 - 2nd International Conference on Engineering Optimization, 6 Sept. - 9. Sept. 2010, Lisbon, Portugal.

Schäfer, B. and Leite, A. C., 2011, "Development Environment for Optimized Locomotion System of Planetary Rovers", Proceedings of the XIV International Symposium on Dynamic Problems of Mechanics - DINAME 2011, São Sebastião, SP, Brazil.

Wong, J. Y., 2001, "Theory of Ground Vehicles", Ed. 3, John Wiley & Sons.

RESPONSIBILITY NOTICE

The authors are the only responsible for the printed material included in this paper.