Zentrum für Technomathematik
Fachbereich 3 – Mathematik und Informatik

Global And Local Optimization Approaches For Launch Vehicles Ascent Trajectory Design

Annalisa Riccardi
Francesco Castellini
Christof Büskens
Michèle Lavagna

Report 11–07

Berichte aus der Technomathematik

Report 11–07 December 2011

GLOBAL AND LOCAL OPTIMIZATION APPROACHES FOR LAUNCH VEHICLES ASCENT TRAJECTORY DESIGN

Annalisa Riccardi
Universität Bremen, Germany, riccardi@math.uni-bremen.de

Francesco Castellini
Politecnico di Milano, Italy, castellini@aero.polimi.it

Prof. Dr. Christof Büskens
Universität Bremen, Germany, bueskens@math.uni-bremen.de

Prof. Dr. Michèle Lavagna
Politecnico di Milano, Italy, lavagna@aero.polimi.it

The paper presents the study on the ascent trajectory optimization problem accomplished during a research activity on Multidisciplinary Design Optimization (MDO) for launch vehicles, undertaken by Universität Bremen and Politecnico di Milano within ESA's PRESTIGE PhD program.

The trajectory optimization problem represents just a part of the overall MDO process when the control variables are treated on the same level of the design variables in a black-box optimization approach. However, given an efficient problem formulation and optimization strategy, it can be inserted as a nested optimization loop in the overall process of design optimization of the entire launch vehicle.

In order to tackle Mixed Integer Non Linear Programming problems required by a MDO framework, several optimization strategies have been integrated: from global and stochastic to local and deterministic, from single to multiobjective.

The description of the optimization strategies is followed by an overview of the ascent trajectory model, constituted of 3-DoF simulation, a phase structure including standard guidance laws for the generation of first guess pitch and yaw profiles, variable thrust, coast phases, and definition of path and final orbit constraints. Results are presented for several test cases (Ariane 5 and VEGA to GTO and LEO orbits), with a comparative analysis of those obtained with global and local optimization approaches, and with different formulations of the problem. Finally, lessons learned on particular modeling aspects that allow improving the problem's smoothness for more efficient and robust local optimization are discussed.

I. INTRODUCTION

Multidisciplinary Design Optimization has been increasingly studied since the 80's in aerospace engineering[1] with the main purpose of reducing monetary and schedule costs for the design of complex airplanes or space vehicles.

Multidisciplinary Design Optimization (MDO) is intended as the coupling together of two or more analysis disciplines with numerical optimization methods. It has been defined by the NASA Langley Research Center (LaRC), Multidisciplinary Design Optimization Branch (MDOB) as *"a methodology for the design of complex engineering systems and subsystems that coherently exploits the synergy of mutually interacting phenomena"*.

The traditional sequential design approach of optimizing each discipline separately, which can lead to sub optimal solution, is substituted by analyzing the interactions between the disciplines and concurrently optimize every subsystems achieving globally optimal design[2,3].

Through the MDO approach in fact, the design space can be more rapidly explored, investigating a high number of possible solutions and obtaining Pareto optimal fronts under different objectives, such as mass, cost, reliability, or mission flexibility. Designers can then select the most promising solutions to be used as good starting points for concepts refinements with more traditional design methodologies.

MDO is a very challenging field of research for both engineers and mathematicians. Exploiting the interaction between the different disciplines, introducing existing or developing new models, solving multiobjective optimization problems with discrete and continuous variables and performing an overall efficient and robust design process are the main issues of a MDO problem for both groups of scientists.

The main obstacle to the successful application of the MDO approach lays in the difficult task of finding a compromise between models simplicity and accuracy, which applies both for the multidisciplinary modeling and at the disciplinary level. This paper focuses on the Trajectory Optimization problem that it is just a crucial part of the overall MDO process, which is described in more details in Ref. 7.

The aim is to perform a simple, fast and robust trajectory optimization to be nested inside the multidisciplinary analysis cycle for improving the performance of the overall optimization. The selected trajectory model, problem formulation and optimization algorithm are analyzed. In particular the paper discusses the following topics:

- Section II: brief overview of the MDO framework developed for the design of Expendable Launch Vehicles.
- Section III: brief overview over the selected optimization strategy for both the MDO problem and the trajectory optimization subproblem.
- Section IV: description of the ascent trajectory optimization problem, from the modeling to its optimization formulation.
- Section V: details of the applicative test case selected for the verification of the results (Ariane 5 to GTO and VEGA to LEO).
- Section VI: model and formulation of the optimization problem enhancements.
- Section VII: critical analysis of the numerical results obtained with the modification proposed in Section VI for the applicative cases presented in Section V.
- Section VIII: conclusion and remarks on the conducted study.

II. FRAMEWORK

The European Space Agency (ESA) proposed in 2009 to co-fund together with the Aerospace Engineering Department of Politecnico di Milano and the Center for Industrial Mathematics of Universität Bremen a joint research in the field of Multidisciplinary Design Optimization (MDO).

Different optimization algorithms, MDO architectures and engineering methods have been developed during the research activity to identify the most suitable for Expendable Launch Vehicles (ELV) design, up to the early preliminary level of detail and considering extensions to more complex applications such as manned and reusable systems[4,5,6].

A research in this field stems from the consideration that, when looking at the future of space exploration, the area with the higher potential for the development of new vehicles is surely that involving space transportation and space launch systems, both for manned and unmanned scenarios.

The engineering modeling of launch systems is a particularly complex task, even restricting the targeted vehicles to classical (i.e. simple cylindrical stages and boosters with no wings) launchers. In the first step of the research described here, the models are kept simple enough to allow execution of a full MDA on a single processor personal computer in a few seconds.

The following disciplinary models have been implemented, making use of freely available external tools or developing new ones from scratch, sequentially executed within the MDA cycle: Propulsion, Geometry, Aerodynamic, Weights, Trajectory, Cost and Reliability for targeting from a conceptual level and up to early preliminary the design of ELV[7].

The straightforward Black Box Optimization MDO architecture has been selected. It consists in an efficient global optimization strategy on top of the multidisciplinary analysis able to explore the entre search space. The MDA takes as inputs the design and disciplinary variables and returns as outputs to the top-level optimizer the design objectives and constraints. The optimizer then recursively calls the model evaluation procedure moving towards feasibility and optimality.

Whereas all other disciplines involve discrete optimization variables, the ascent trajectory dynamics is governed by continuous control parameters, hence an optimal solution can be found employing efficient local optimization techniques. For this reason, in the presented framework, the paper will focus on the research developed for designing a robust model and selecting an efficient optimization strategy for the ascent trajectory optimization sub-problem, to include as a nested loop within the MDA cycle.

III. THE OPTIMIZATION STRATEGIES

In order to tackle Mixed Integer Non Linear Programming problems required by a MDO framework, several optimization strategies have been integrated: from global and stochastic to local and deterministic, from single to multiobjective.

Global Stochastic Multi Objective approach is based on the collaborative hybridization of three different Evolutionary Algorithms: Non-Dominated Sorting Genetic Algorithm (NSGA-II)[8], Double Grid Multi Objective Particle Swarm Optimization (DGMOPSO)[9], and Multi Objective Ant Colony Optimization for continuous domains (MOACOr)[10]. The selection of the algorithms has been based on their efficiency, in terms of function evaluations required for convergence, and robustness on a set of benchmark mathematical problems[9]. The idea of the hybrid algorithm is to steer the algorithm toward the strategy that achieved the best results, in terms of contribution to the current Pareto Front, in the previous iteration. The initial population is equally divided in three groups and each group evolves with one of the evolutionary techniques. The results are then recombined, the fast non-dominated sorting operator of NSGA-II algorithm is applied to the joint population and the ordered non-dominated fronts are added to the archive until the maximum size is reached. Two hybridization coefficients are employed to steady the hybrid algorithm toward the evolutionary strategy that is achieving the best results: the percentage of population that evolves with each algorithm is proportional to the percentage of individuals in the current archive that derive from it[5].

Global Stochastic Single Objective approach: is based on the original Particle Swarm Optimization (PSO)[11]. It has been shown to be generally more efficient than the more traditional single-objective genetic algorithms.

Local Determinist Single Objective approach is a combined SQP (Sequential quadratic programming) and primal-dual IP (Interior-Point) method, which was designed to solve sparse large-scale NLP problems with more than hundreds of millions of variables and constraints. It has been developed in the library WORHP ("We Optimize Really Huge Problems") by the joint work of the teams from the University of Bremen and the team from the University of Würzburg[12,13]. Its robustness was proved by the CUTEr test set, consisting of 920 sparse large-scale and small dense problems, of which WORHP is able to solve 915. Moreover WORHP successfully solved several space application problems, e.g. reentry, ascent and low thrust trajectory optimization problems.

IV. THE ASCENT TRAJECTORY OPTIMIZATION PROBLEM

The trajectory and guidance block can be considered at a lower level in a MDO nested optimization loops (NOL) as already mentioned in the previous paragraph. The trajectory optimization variables are not shared with other subsystems so the variables that define the design of the launcher, needed for the computation of the equation of motion, can be frozen at system level and a nested optimization loop can be performed instead that a simulation during the multidisciplinary analysis. The advantages of this approach are mainly two:

- the number of the optimization variables of the system level MDO problem decrease substantially
- The trajectory optimization variables are continuous and if a good first guess is available a more efficient local optimization approach can be employed.

On the other hand a fast and robust trajectory optimization has to be performed at each evaluation of the MDA.

Dynamics and guidance models

Three Degrees of Freedom (DoF) dynamics and limited environmental models (rotating Earth with zero-order gravity model, US 76 atmosphere, no wind), are considered appropriate for the level of accuracy required in a MDO context.

The trajectory is integrated from launch to orbit insertion with a Runge-Kutta-Fehlberg 45 algorithm. The rotational dynamics is neglected, implicitly assuming that the launcher is capable of providing the necessary pitch and yaw profile without steering losses. A static controllability verification is however performed on the basis of the CoG position and pitching moment coefficient. Moreover, in case of non axial-symmetric launchers the roll angle is fixed (velocity and longitudinal axis in the symmetry plane), so that the aerodynamic coefficients are function only of Mach number and total angle of attack. Complementary models include the evaluation of the aero-thermal loads at stagnation point, structural loads (dynamic pressure, axial and lateral accelerations), thrust and Isp variation due to altitude, and static controllability verification.

Optimization problem

Parameterized pitch and yaw angles as well as thrust throttling are used as control variables, while a scaling factor for the reference payload mass is included among the optimization variables and also represents the maximization cost function in a performance based optimization. The last stage burnout can be activated as additional optimization variable and the ignition time for an orbital circularization burn complete the set of trajectory design variables, allowing to reach high circular or moderately eccentric orbits.

Constraints are imposed on the final orbital parameters as well on the path constraints: maximum dynamic pressure, maximum heat flux before payload fairing jettison, maximum axial acceleration, maximum angle of attack, static controllability (available thrust torque larger than aerodynamic moment).

The first guess is provided by standard guidance laws, composed of vertical launch, exponential pitch push-over, gravity turn and bilinear tangent law for pitch and target inclination for yaw. Pitch and yaw profiles are obtained with linear interpolation of the nodal values in a limited range around the first guess solution. In this way local optimization processes are started with a reasonable first guess (i.e. a "flying" trajectory rather than one ending in a crash on the planet), allowing for fast convergence to the final optimum.

With direct methods the optimal control problem is transformed in a nonlinear problem of the form

$$\min_{(x_1,\dots,x_n)\in\Re^n} \; -x_1$$
$$\text{subject to } l_i \le x_i \le u_i;\ for\ i=1,\dots,n$$
$$g_i(x_1,\dots,x_n)\le 0;\ for\ i=1,\dots,m$$

where n is the total number of optimization variables that depends on the discretization used for the controls and the launcher configuration, x_1 is the payload scaling factor, l_i and u_i are respectively the lower and the upper bound of the box constraints of the optimization variable x_i and g_i the above mentioned constraints on final target orbit and path constraints.

In the implementation of the optimization problem the design and control variables are scaled between 0 and 1 as well as the constraints values that are scaled with their reference values.

Three main figures of merit of the local trajectory optimization process have been considered:

- ***Robustness*** in terms of repeatability of the results. This can be evaluated mainly in two ways: varying the first guess, the algorithm parameters (in case of deterministic strategies) or the seed (in case of stochastic strategies) and verifying that the same or comparable optimal solutions are obtained; or varying the launch vehicle's design parameters and verifying that the sensitivity of the optimal payload performance matches what physically expectable. The easiest way is to vary the upper stage's dry mass and verify that the same variation in payload is achieved.
- ***Performance*** in terms of optimal value and constraints violation.
- ***Computational efficiency***: average time required to reach the optimal solution.

Although all these aspects are extremely important for a successful trajectory optimization tool, robustness seems the most critical within MDO problems. In fact, a trajectory framework that does not allow for reliable evaluation of the payload performance may artificially bias the multidisciplinary optimization towards given design solutions, for which the trajectory optimization proceeds better.

The ascent trajectory has been optimized with both the global PSO and the local WORHP algorithms for the Ariane5 ECA to GTO and VEGA to LEO as test cases. The use of the available multiobjective strategies has been considered inefficient for the kind of problem.

A comparison of the performance of the two algorithms on the provided model led to further improvements on its robustness and regularity presented in the flowing sections.

V. APPLICATIVE TEST PROBLEMS

The optimization problems for Ariane 5 ECA and VEGA slightly differ in terms of variables and flight phases.

The trajectory model for the Ariane5 ECA to GTO launched from Kourou, is divided in 5 phases: vertical take off, pitch over maneuver (constant optimized yaw and linear pitch followed by exponential decay of angle of attack until gravity turn condition is met), first stage flight with boosters (pitch follows the gravity turn and yaw the target inclination), first stage flight without boosters (pitch and yaw are both optimized), second stage flight (pitch follows the bilinear tangent law and yaw is optimized). The throttle of the liquid engines is constant at 100% and the solid boosters have a simplified two-level thrust profile. Hence, only trajectory optimization variables related to payload mass and pitch and yaw profiles are used, for a total of 11 continuous variables listed in Table 1 with relative bounds.

Var	Description	LB	UB
X_{PL}	Payload scaling factor [-]	0.5	1.5
$\Delta\psi_{PO}$	Launch azimuth, in terms of yaw deviation with respect to the target inclination law [deg]	-10	10
$\Delta\Psi_{EPC}$	Yaw deviation with respect to the target inclination law during EPC's flight (no boosters) [deg]	-10	10
$\Delta\Psi_{ECA}$	Yaw deviation with respect to the target inclination law during ECA's flight [deg]	-10	10
$\Delta\theta_{PO}$	Maximum pitch-over angle [deg]	1	5
t_{PO}	Pitch-over duration [s]	2	10
$t_{PO,decay}$	Pitch-over decay time [s]	1	5
$\Delta\theta_{EPC}$	Pitch deviation with respect to gravity turn during EPC's flight (no boosters) [deg]	-10	20
$\Delta\theta_{BTL,i}$	Pitch discontinuity (with respect to the pitch at the end of EPC's flight) at the beginning of the bilinear tangent law phase for upper stage's pitch [deg]	-50	50
$\Delta\theta_{BTL,f}$	Pitch value at the end of the bilinear tangent law for upper stage's pitch [deg]	0	50
ζ_{BTL}	Bilinear tangent law's shape parameter (ζ=0 → linear, ζ>0: super-linear, ζ<0: sub-linear) [-]	-1	1

Table 1 Description of the optimization variables and their bounds for the complete considered search space of the trajectory optimization problem for Ariane 5 ECA's flight to GTO

All constraints are active with reference values reported in Table 1. An error of 10 km on the final semiaxis, 0.01 on the eccentricity and 0.5 degree on the inclination are allowed.

	Lower Bound	Upper Bound
Target semiaxis (Km)	24475-10	24475+10
Target eccentricity (-)	0.7292-0.01	0.7292+0.01
Target inclination (deg)	6-0.5	6+0.5
Maximum axial acceleration (g)	-∞	4.55
Maximum heat flux before payload fairing jettison (MW/m^2)	-∞	30
Maximum dynamic pressure (Pa)	-∞	57000
Controllability violation	-∞	1.5
Maximum angle of attack (deg)	-∞	5

Table 2 Ariane 5 trajectory constraints bounds

VEGA's ascent to a polar LEO from Kourou is divided in 9 phases each with different guidance laws for pitch and yaw: vertical take off, pitch over maneuver (constant optimized yaw and linear pitch followed by exponential decay of angle of attack until gravity turn condition is met), first and second stage flight (pitch follows the gravity turn and yaw the target inclination), third stage flight (optimized pitch and yaw), fourth stage flight (pitch follows the bilinear tangent law and yaw is optimized), coast phase between Z23 and Z9 flights and coast phase of the upper stage until circularization burn with pitch and yaw tangential to the velocity and null thrust, circularization burn with upper stage at full thrust. As in the previous case the trajectory optimization variables are related to payload mass, pitch and yaw profiles, with the addition of the coast times and circularization burn time, for a total of 15 variables listed in Table 3.

Var	Description	LB	UB
X_{PL}	Payload scaling factor [-]	0.5	1.5
$\Delta\psi_{PO}$	Launch azimuth, in terms of yaw deviation with respect to the target inclination law [deg]	-10	10
$\Delta\Psi_{Z23}$	Yaw deviation with respect to the target inclination law during Zefiro-23's flight [deg]	-10	10
$\Delta\Psi_{Z9}$	Yaw deviation with respect to the target inclination law during Zefiro-9's flight [deg]	-10	10
$\Delta\Psi_{AVUM}$	Yaw deviation with respect to the target inclination law during AVUM's flight [deg]	-10	10
$\Delta\theta_{PO}$	Maximum pitch-over angle [deg]	1	5
t_{PO}	Pitch-over duration [s]	2	10
$t_{PO,decay}$	Pitch-over decay time [s]	1	5
$\Delta\theta_{Z23}$	Pitch deviation with respect to gravity turn during Zefiro-23's flight [deg]	-10	20
$\Delta\theta_{Z9}$	Pitch deviation with respect to gravity turn during Zefiro-9's flight [deg]	-10	20
$\Delta\theta_{BTL,i}$	Pitch discontinuity (with respect to the pitch at the end of EPC's flight) at the beginning of the bilinear tangent law phase for upper stage's pitch [deg]	-50	50
$\Delta\theta_{BTL,f}$	Pitch value at the end of the bilinear tangent law for upper stage's pitch [deg]	-5	50
ζ_{BTL}	Bilinear tangent law's shape parameter ($\zeta=0$ → linear, $\zeta>0$: super-linear, $\zeta<0$: sub-linear) [-]	-1	1
$t_{coast,Z}$	Duration of coast phase between jettison of Zefiro-23 and ignition of Zefiro-9 [s]	30	100
$t_{CB,ign}$	Circularization burn ignition deviation with respect to ($t_{apocentre}-0.5\cdot t_{CB}$), in percentage of the total circularization burn time t_{CB} [-]	-0.1	0.1

Table 3 Description of the optimization variables and their bounds for the complete search space of the trajectory optimization problem for VEGA's flight to polar LEO

	Lower Bound	Upper Bound
Target semiaxis (Km)	7878-10	7878+10
Target eccentricity (-)	0.0-0.01	0.02+0.01
Target inclination (deg)	90-0.5	90+0.5
Maximum axial acceleration (g)	-∞	7.5
Maximum heat flux before payload fairing jettison (MW/m^2)	-∞	30
Maximum dynamic pressure (Pa)	-∞	57000
Controllability violation	-∞	1.5
Maximum angle of attack (deg)	-∞	5

Table 4 VEGA trajectory constraints bounds

Also in the VEGA test case all constraints are active with reference values reported in Table 4. The tolerances on the orbital parameters are the same as in the Ariane settings.

VI. MODEL AND OPTIMIZATION PROBLEM ENHANCEMENT

Global and local trajectory optimization processes have been performed identifying two main features of the model and the optimization problem formulation that largely influence the behaviour of the algorithms. In particular

- **Model**: stopping criteria on the fulfilment of the target orbit constraints.
- **Problem formulation**: use of equality or inequality constraints for the matching of the target orbit.

Finally in this section a sensitivity analysis on the effect of the number and bounds of the optimization variables is discussed.

Model enhancements

Although the regularity of the objective and constraints functions is not an issue when dealing with stochastic global optimization algorithms, particular attention has to be paid to this aspect when applying gradient-based methods such as WORHP.

While the objective is a linear function, therefore smooth by nature, the constraints are non linear functions. In particular, the final orbit constraints were found to be extremely non smooth with the models developed for the PSO trajectory optimization case. This issue was highlighted from the very poor robustness properties initially shown by the trajectory models when applying a local method. In particular, very small variations in any of the launcher design parameters, WORHP's settings or first guess would result in a completely different solution being found, indicating the presence of a large number of local minima with very small regions of attraction. Path constraints on the other hand didn't result an issue for the optimization problem, probably due to their dependency on a smaller set of optimization variables.

The problem was identified by plotting the constraints surfaces as a function of two of the most influential optimization variables: the initial pitch-over angle and the payload scaling factor (which is also the optimization's objective function) freezing the remaining optimization variables to the optimal values returned by WORHP and running different simulation on the grid nodes.

The main issue highlighted by this analysis is shown in Figure 1for a flight of Ariane 5 ECA towards a standard GTO. The final orbit's semiaxis on the z-axis presents a flat region where the target semiaxis is matched. Since the integration of the equation of motion is stopped as soon as the required orbital energy is reached, this region is large but extremely "bumpy". In fact, there are hundreds of local minima and maxima of the constraint surface in the feasibility area, due to the discrete nature of variable step size integration process. This results in different instants of integration stopping, sometimes within and sometimes out of the allowed tolerance on semiaxis.

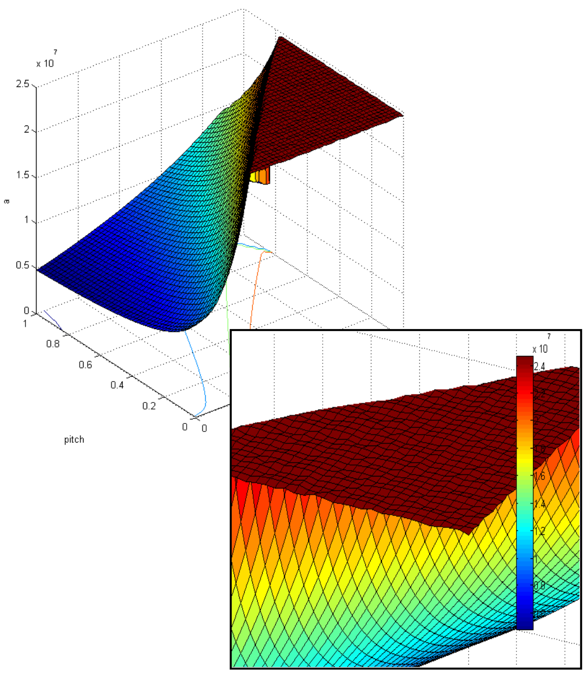

Figure 1 Final orbit's semiaxis constraint surface as a function of initial pitch-over angle and payload mass for Ariane 5 ECA to GTO. The plot is obtained by freezing all other optimization variables.

Such a model leads to a disconnected feasible region. For different initial guess or varying the algorithm parameters WORHP gets stuck in different local minima as shown in Figure 2 where are visualized the contour lines of the semimajor axis function and the corresponding disconnected feasible area.

The set of feasible solutions is therefore disconnected, so that each time the optimization is started with different settings, the final solution can end up being anywhere in the feasibility region without a guarantee on reaching the best feasible point (as it happen in Figure 2 where the optimal solution returned by WORHP is represented by the red cross in the feasible space).

While the presented model is good for global stochastic optimization approach, since the feasible area is wider, the model has to be redefined for local gradient based algorithms. It can be corrected by simply avoiding the stopping of the integration at the target orbital energy, imposing that the propellant of the upper stage is always fully depleted. It implies a reduction of the feasible region dimensions, so inefficient for global strategies but allows obtaining a smooth model with no local optima (Figure 3). Hence, WORHP is capable of pushing the solution as further right as possible, so that the achieved optimum (i.e. red cross) corresponds to the highest payload solution in the feasible set.

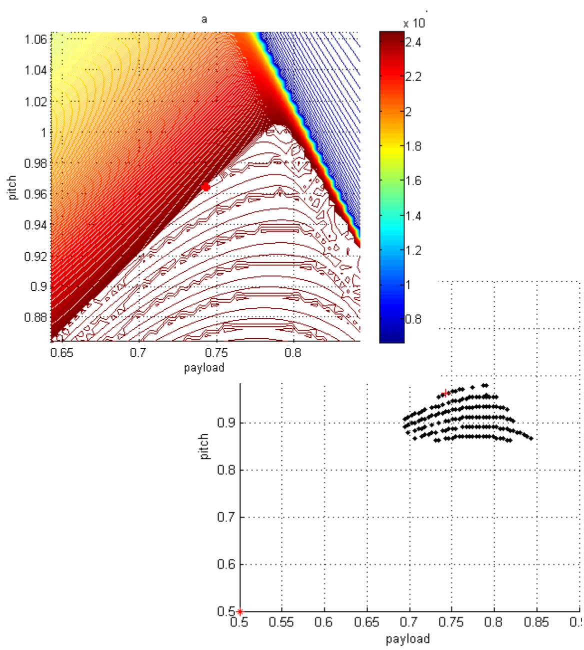

Figure 2 Top: contour plot of the final semiaxis constraint surface. Bottom: set of feasible solutions obtained varying initial pitch-over and payload. The red solution at x=y=0.5 is the first guess, the red solution in the feasibility region is the solution obtained by WORHP, clearly a local optimum.

To recover the wider feasible region found with the stopping criteria and maintain the regularity of the problem an additional optimization variable can be added to the problem. The additional variable models the remaining propellant in the last stage on reaching the target orbit, allowing variable time of flight. Naturally the optimum still relies in the solution with no fuel left over but the feasible search space has been enlarged to the original one (Figure 4). The figure refers to a coarser grid over the search space since a complete optimization must be performed at each grid node though it gives anyway a taste of the behaviour.

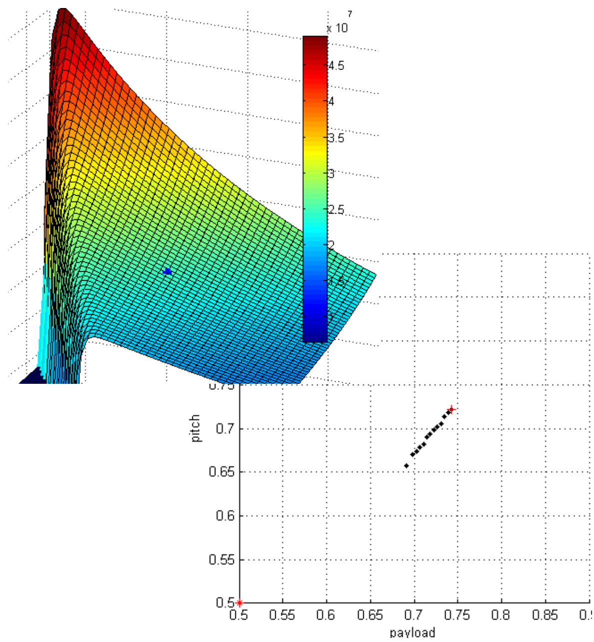

Figure 3 Top: 3D plot of the semiaxis constraint surface for Ariane 5's flight to GTO after the removal of the integration stopping condition. Bottom: set of feasible points, first guess at x=y=0.5 and globally optimal solution returned by WORHP.

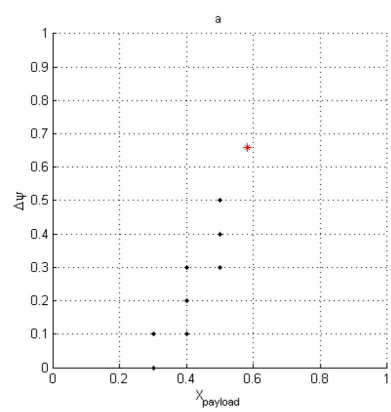

Figure 4 set of feasible solutions obtained varying initial pitch-over and payload and leaving optimizable the final time of flight.

Optimization problem formulation enhancements

The distinction between equality an inequality constraints has been another major issue registered in the original trajectory model designed for the PSO algorithm where all constraints are treated as inequality constraints allowing given tolerances on the final orbital parameters. This constraints relaxation is necessary in case of a global optimization approach, too coarse to allow the precise matching of equality constraints. However, WORHP is capable of handling both equality and inequality constraints, and the choice among the two types does influence the behaviour of the optimization algorithm. Treating a equality constraints as inequality enlarge the corridor of the acceptable solutions adding to the tolerance set by the user the tolerance employed by the algorithm for feasibility determination.

In particular, inequality (g_i, i=1,..,3) and equality (h_i, i=1,..,3) constraints on the final target orbit are defined as follow

$$g_1(x) = \frac{a(x) - a_{target}}{a_{tol}}; \; h_1(x) = \frac{a(x) - a_{target}}{a_{norm}};$$

$$g_2(x) = \frac{e(x) - e_{target}}{e_{tol}}; \; h_2(x) = \frac{e(x) - e_{target}}{e_{norm}};$$

$$g_3(x) = \frac{i(x) - i_{target}}{i_{tol}}; \; h_3(x) = \frac{i(x) - i_{target}}{i_{norm}};$$

Where x is the vector of the optimization variables, a_{target}, e_{target}, i_{target} are the target orbit semiaxis, eccentricity and inclination, a_{tol}, e_{tol}, i_{tol} are the tolerance allowed by the user and a_{norm}, e_{norm}, i_{norm} are the scaling factors and the constraints bounds are defined as

$$g_i(x) \le 0; \quad h_i(x) = 0; \quad for\,all \;\; i = 1,...,3.$$

On top of these definitions, WORHP further defines a tolerance on the constraints values, corresponding to the parameter *TolFeas*. In order to ensure a fair comparison between the inequality and equality constraints models, the following relation must therefore hold

$$a_{norm} = \frac{a_{tol}}{TolFeas}; \; e_{norm} = \frac{e_{tol}}{TolFeas}; \; i_{norm} = \frac{i_{tol}}{TolFeas};$$

so that when WORHP considers acceptable the constraints violation on a given parameter in the equality case, the error is lower or equal to the allowed tolerance.

The last aspect that has been investigated is the definition of the number and bounds of the control parameters listed in Table 1 and Table 3 to avoid redundant optimization variables and increase the optimization efficiency. In general, the following fact applies:

- If the search space is constrained too much (small number of control variables with narrow bounds) than the achieved optimal solutions may be largely underestimating the payload mass.
- If the search space is enlarged too much (large number of control variables with wider bounds) than the computational times may become excessive and the search space may become extremely multi-modal, reducing the robustness of the local search.

In order to find a balanced compromise of these two approaches, an analysis on the effect of the number and bounds of the control variables on the optimization process has been carried out. Particular attention has been paid in this case to the performance and robustness allowed by the different search space dimensions.

Numerical results of the applicative test cases presented in V are presented in the next section as a proof of the topics stated above.

VII. NUMERICAL RESULTS

Using the actual launcher design parameters, trajectories for Ariane 5 and VEGA have been optimized with both the global PSO, performing 5 runs of 500 iterations and 250 particles for stochastic effects, and the local WORHP with the model and problem formulation enhancements discussed in the previous section. The first guess: is provided both from standard guidance laws and from fast global optimization runs with the PSO algorithm (for example 10 particles and 10 iterations).

The results in Table 5, refers to the best available solutions, in terms of payload mass starting from the actual configuration of Ariane and VEGA and from the design given by the multidisciplinary analysis, taking as reference value of payload mass for Ariane 5, 10050 Kg, and for VEGA 1500 Kg.

	PSO best	PSO stdev	WORHP
Ariane 5 actual	10891	1.0%	10944
Ariane 5 MDA	12482	0.1%	12693
VEGA actual	1714	0.5%	1715
VEGA MDA	1408	0.1%	1403

Table 5 Trajectory optimization results for Ariane 5 ECA and VEGA test cases, payload mass values in kg, obtained with PSO and WORHP for launcher parameters frozen to the actual design values and computed by the analysis.

The optimization performed with the two optimization strategies leads to consistent results. Moreover note that the stochastic effects do not lead to excessive standard deviations in the payload performance among the different PSO runs. Although this is obtained in a large number of model evaluations reflected in long computational times corresponding to about 4 and 4.5 hours for Ariane and VEGA on a single 2.10 GHz processor. An OpenMP parallelization of the stochastic code can halved the computational time using standard dual core pc. The stochastic approach is still not comparable, in terms of computational efficiency, to the performance of the local algorithm that is able to perform a complete optimization process in the order of 5 to 20 minutes.

The trajectory models tend to slightly overestimate the payload mass, specifically by 8.9% for Ariane and 14% for VEGA for fixed design input variables while in case the design is given by the MDA a bigger overestimation of 26% for Ariane and on the contrary a small underestimation of 6% for VEGA. The optimistic evaluation of the payload mass can be addressed to the neglecting of steering losses throughout the ascent phase and the assumption of constant specific impulse. Introduction of a steering ΔV in the propellant budget as well as of an empirical evaluation of the effect of throat erosion, particularly relevant in case of SP motors, may therefore allow improving the payload assessment accuracy. The over/underestimation in the MDA case is consistent with the inaccuracy in the model, as discussed in [6].

Equality vs Inequality constraints definition

The use of equality or inequality constraints for local optimization strategies to verify which of the two definitions gives the best compromise of performance, efficiency and robustness of the optimization process, for the Ariane 5 ECA and VEGA (both actual design and MDA) cases have been analyzed. The smallest possible number of control parameters has been used to reduce the influence of this aspect (see next paragraph). All pitch-over and upper stage's bilinear tangent law parameters have therefore been frozen to reasonable values, and it has been allowed to the optimizer to vary only the pitch and yaw values during the lower stages flight.

For all the analyzed cases, 4 runs have been executed: three starting from initial guess given by a fast run of PSO (10 particles and 10 iterations) and one from standard guidance law. In terms of computational efficiency the results are comparable. Rather interesting are the results on the performance of the algorithm and robustness of the model. In Table 6 the results related to the maximum value of payload mass taken from the different run of the local algorithm are listed comparing the case of inequality and equality constraints definition for both actual and MDA design.

	Payload mass [kg]	
	Eq	Ineq
Ariane 5 actual	10810	10823
Ariane 5 MDA	12663	12693
VEGA actual	1506	1521
VEGA MDA	1394	1403

Table 6 Study on equality vs inequality constraints: Performance, for the optimization process on different test cases.

Inequality constraints definition allows obtaining higher payload masses, typically in the 1-3 % range. This is justified by the more precise matching of the final orbital parameters in case of inequality constraints (Figure 5), which results in a small penalty in the objective and an increase in the algorithm performance.

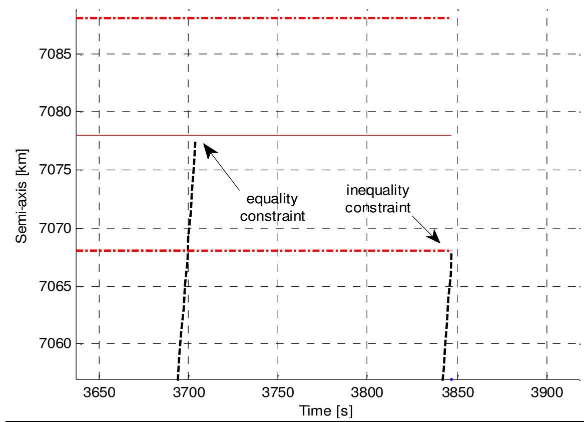

Figure 5 Inequality vs. equality constraints matching example: final semiaxis values for a trajectory of VEGA towards a circular polar 700 km LEO with circularization burn.

The study on the robustness of the algorithm subject to change in the constraints definition is performed in two different ways: first considering the standard deviation σ on the payload mass values, second evaluating the sensitivity of the payload mass to the upper stage's dry mass (Table 7).

	Payload mass σ [%]		$\partial M_{PL}/\partial M_{US}$ [%]	
	Eq	Ineq	Eq	Ineq
Ariane 5 actual	0.07	0.19	-0.978	-0.989
Ariane 5 MDA	0.04	1.31	-0.982	-1.151
VEGA actual	1.39	1.54	+1.032	-0.964
VEGA MDA	2.94	4.16	+1.018	-0.780

Table 7 Study on equality vs inequality constraints: robustness, for the optimization process on different test cases.

The values refer on the results obtained in the four different runs. Equality constraints definition allows much more robust optimization process, both in terms of standard deviation of the estimated payload mass and sensitivity to the variation of the launcher's design (upper stage's mass).

It is therefore suggested to start optimizations from the equality constraints definition, and try the inequality definition case afterwards, only if stretching the performance by a few percent is required.

Problem dimension study

The analysis on the effect of the number and bounds of the control parameters has been performed for Ariane 5 and VEGA only for the actual design case. The procedure is to start the local optimization search with the smallest search space (most of the control variables frozen) and gradually enlarge it to include more control variables and/or wider bounds, continuing to monitor the behaviour of the optimization and assess the optimization's performance and robustness properties.

Only the most relevant of the attempted configurations in terms of achieved results (i.e. performance, robustness and efficiency) are detailed here. Each configuration constitutes a different local optimization problem, solved with WORHP with the equality constraints definition (which allows for better robustness as described in the previous paragraph). Optimization runs with inequality constraints were also attempted, with the purpose of verifying the increase in payload performance related to the relaxation of the constraints and labelled with *Configuration x_IN*. *Configuration 0* refers to the complete one.

In Table 8 and Table 9 are listed the analyzed configurations for the two test cases and in Table 10 and Table 11 the best optimal solutions returned by WORHP for the corresponding configurations. Four runs have been executed, three runs starting from the fist guess given by PSO and one with standard guidance laws. The not optimized variables are constrained to value 0 apart from $\Delta\theta_{PO}$=2, t_{PO}=6 and $t_{PO,decay}$=2 for the Ariane case and $\Delta\theta_{PO}$=4, t_{PO}=6, $t_{PO,decay}$=2 and t_{coast}=65 (angles are in degrees and time in seconds) for VEGA.

Conf	Description
A	All optimization variables frozen, except X_{PL}, $\Delta\Psi_{EPC}$, $\Delta\theta_{EPC}$
B	Same as A, adding $\Delta\theta_{PO}$
C	Same as B, adding $\Delta\theta_{BTL,f}$
D	Same as C, adding $\Delta\theta_{BTL,i}$
E	Same as D, adding ζ_{BTL}

Table 8 Analyzed configurations description for the Ariane 5 to GTO case.

Conf	Description
A	All optimization variables frozen, except PLSF, $\Delta\Psi_{EPC}$, $\Delta\theta_{EPC}$
B	Same as A, adding $\Delta\theta_{PO}$ for better balance of drag and gravity losses
C	Same as B, adding $\Delta\Psi_{Z23}$ for better inclination matching
D	Same as C, adding the coast time between Zefiro-23 and Zefiro-9 stages
E	Same as D, adding pitch during Zefiro-23's flight
F	Same as E, adding the linear pitch-over time

Table 9 Analyzed configurations description for the VEGA to LEO case.

Conf	nVars	Payload	Robustness	Efficiency
A	3	8919	0	<0.5 min
B	4	9936	0	<0.5 min
C	5	10568	1.10%	<1 min
D	6	10852	0	<1 min
E	7	10878	0	<2 min
E_IN	7	10917	4.25%	<2 min
0	11	10889	0.96%	<10 min
0_IN	11	10944	0.62%	<10min

Table 10 Summary of optimization results for the different configurations of Table 8. Results are reported in terms of performance (payload mass [kg]), robustness (σ from the different initial guesses) and efficiency (CPU times).

Conf	nVars	Payload	Robustness	Efficiency
A	3	1372	0	<1 min
B	4	1642	0	<1 min
B_IN	4	1659	0	<1 min
C	5	1657	0	<1 min
D	6	1675	0	<1 min
E	7	1682	0.52%	<2 min
F	8	1699	0.21%	<3 min
0	15	1706	0.17%	<12 min
0_IN	15	1715	0.51%	<25 min

Table 11 Summary of optimization results for the different configurations of Table 9. Results are reported in terms of performance (payload mass [kg]), robustness (σ from the different initial guesses) and efficiency (CPU times).

Several important general conclusions are drawn from the analysis of the optimization results.

Standard guidance laws with default parameters (i.e. in the middle of the optimization bounds) often result in failure of the local optimization process or extremely long computational times. In some cases, the standard laws provide a reasonable enough first guess to allow for successful optimization, and it is also possible to manually tune the optimization bounds to achieve such a solution. However, this approach is not suggested: the execution of a fast global optimization run only takes a few seconds and allows to obtain a good initial trajectory (i.e. close to feasibility), from which WORHP is in most cases capable of reaching the optimal solution.

For each configuration, several runs with smaller bounds for the different optimization variables were attempted. This however never resulted in significant advantages in terms of robustness or efficiency of the optimization and suggests that the better approach is to keep the bounds of the optimization variables rather large (within reason, such as those reported in Table 1 and Table 3), in order to avoid losing potentially interesting regions of the search space.

The initial pitch-over phase is particularly critical, due to the high sensitivity of the final state to the pitch profile at the beginning of the flight. The introduction in the optimization problem of at least one of the pitch-over parameters allows to greatly increasing the payload performance as well as the maintenance of a very good robustness. However, the further addition of t_{PO} and $t_{PO,decay}$ for the Ariane case results in a negligible improvement of the performance (<0.05%). On the contrary, the enlargement of the search space causes difficulties in the local search, sensibly reducing the robustness of the process probably due to the generation of local optima. This behaviour is justified by the fact that the three parameters are strictly related, and a modification in one variable can be partially or totally compensated with an opposite modification of another. The launch vehicle's pitch-over rate (or t_{PO}) and time to gravity turn (or $t_{PO,decay}$) are typically used as independent variables in ascent trajectory optimization. Nevertheless, the results obtained with WORHP for Ariane suggest that t_{PO} and $t_{PO,decay}$ can safely be fixed to reasonable values, leaving only $\Delta\theta_{PO}$ as degree of freedom to tune the trajectory's pitch-over, with no relevant penalty in payload and a significant advantage in robustness and computational times. On the contrary for the VEGA test case, the introduction of the pitch-over time t_{PO} causes a non negligible improvement. In fact, for VEGA's flight to circular LEO, a strong pitch-over enhances the performance. Since a 5 deg upper bound for the AoA is imposed, reaching this value with a faster rotation allows to gain further payload mass. Hence, for all those cases when the upper (or lower) bound of the pitch-over angle is reached, it is suggested to also optimize the rotation's duration, whereas the pitch-over decay time can still be frozen.

The second largest improvement in performance for the trajectory of the Ariane was achieved when introducing in the optimization problem the bilinear tangent law parameters for the upper stage's flight. In particular, the addition of the pitch value at orbital insertion ($\theta_{BTL,f}$, Conf. C) determined payload values with the equality and inequality constraints definition respectively 3% and 1% lower than the reference performance from PSO runs. This was achieved with higher standard deviations and computational times, which are nevertheless justified by the relevant performance gain. Also the introduction of the pitch discontinuity at upper stage's ignition ($\Delta\theta_{BTL,i}$, Conf. D) and of the bilinear tangent law's shape parameter (ζ_{BTL}, Conf. E) caused performance improvements, though of smaller entity. On the contrary for VEGA the introduction of optimized pitch profile during the upper stage's burn does not largely affect the optimization process. In fact, the bilinear tangent law profile is only active for the first part of the upper stage's flight, before the engine's cut-off at the achievement of the required orbital apocentre. During the remaining part of the upper stage's flight, the longitudinal axis is maintained tangential to the velocity, both during the coast phase (not relevant) and the circularization burn phase. The attempts to introduce one or more of the bilinear tangent law's parameters resulted in no relevant improvement in performance

For the VEGA test case the following remarks state: the coast phase duration introduces one more degree of freedom into the problem, and therefore allows improving the performance by about 1% while circularization burn ignition time has a very small sensitivity on the overall performance and robustness of the problem.

In both cases the complete configurations lead to the best performance at the price of loosing computational efficiency and lack in robustness. The full search space exploitation is not recommended at least for the first attempt. It is instead in general suggested to freeze the variables that results in minor performance enhancement. Summarizing:

- For Ariane 5: freeze two of the pitch-over parameters, the pitch-over direction, and the yaw deviations for stages except for the first exo-atmospheric component, as in Configuration E
- For VEGA: freeze one of the pitch-over parameters, two of the yaw deviations, all bilinear tangent law parameters and the circularization burn ignition time as in Configuration F

VIII. CONCLUSIONS

The paper focuses on a critical analysis of the ascent trajectory model, optimization problem formulation and optimization strategies. The goal is to achieve a fast, consistent and robust performance estimation to be inserted as nested optimization loop within the multidisciplinary analysis for a MDO process.

Among the single objective optimization strategies, the deterministic gradient based optimization algorithm implemented in WORHP overtakes the stochastic approach of the PSO algorithm in terms of computational efficiency. Whereas the evolutionary algorithm does not need particular assumptions on the model smoothness and on the given first guess, WORHP is strict related to the regularity of the model and to the initial guess. To improve the behaviour of WORHP, two key points were developed: 1) regularity issues in the model that were leading to disconnected optimal fronts were solved, and 2) very fast initial PSO runs were introduced to provide initial guesses close to feasibility. These enhancements allowed WORHP to overtake the PSO algorithm also in terms of performance. The robustness of the local approach was reached with a reformulation of the problem constraints as equality constraints. The robustness of the stochastic algorithm was instead proven by running different trajectory optimizations with different seeds for populating the initial set of particles.

Finally, a sensitivity analysis over the set of optimization variables and their bounds, supported by the conclusions above, enabled to reduce the number of relevant optimization variables, reaching an acceptable balance between performances and robustness.

IX. ACKNOWLEDGEMENTS

The authors would like to thank Guillermo Ortega, Antonio Rinalducci, Alvaro Martinez Barrio and Sven Erb of ESA TEC-ECM for the continuing support in the research effort as well as Javier Ventura-Traveset and Carlos Lopez from ESA Education Office for setting up the prestige initiative, the WORHP team for the use of the optimization libraries, Prof. Kalyanmoy Deb and Abel Garcia Najera for providing the source code of the NSGA-II and MOACOr algorithms, and Chris Laurel for the support in all software related issues. advices in the models set-up and evaluation from ESA and VKI experts have also been greatly appreciated, in particular from: Jens Kauffmann, Carlos Corral Van Damme, Gianluigi Baldesi, Marco De Rosa, Francesco di Matteo, Marco Chiappone, Michel Van Pelt, Cem Ozam, and Patrick Rambaud.

[1] Sobieszczanski-Sobieski J., "Multidisciplinary Optimization for Engineering Systems: Achievements and Potential", NASA TM 101566, March 1989

[2] Olds J.R., "The suitability of selected multidisciplinary design and optimization techniques to conceptual aerospace vehicle design", 4th AIAA/USAF/NASA/OAI Symposium on Multidisciplinary Design and Optimization, Cleveland, Ohio, September 1992.

[3] Braun R.D., Moore A.A., Kroo I.M., and Stanley D.O., "Use of the collaborative optimization architecture for launch vehicle design", 6th AIAA/USAF/NASA/ OAI Symposium on Multidisciplinary Design and Optimization, Bellevue, WA, September 1996.

[4] Castellini F., Riccardi A., Lavagna M., and Bueskens C., "A First Step Towards Svago: The Space Vehicles Analysis And Global Optimization MDO Tool", 4-th International Conference on Astrodynamics Tools and Techniques, Madrid, Spain, May 2010.

[5] Castellini F., Riccardi A., Lavagna M., and Bueskens C., "Multidisciplinary Design Optimization models and algorithms for space launch vehicles", AIAA/ATIO/ISSMO Multidisciplinary Analysis and Optimization Conference, Fort Worth, TX, September 2010.

[6] Castellini F., Riccardi A., Lavagna M., and Bueskens C., "Global and Local Multidisciplinary Design Optimization of Expendable Launch Vehicles", AIAA Multidisciplinary Design Optimization Specialists Conference, Denver, CO, April 2011.

[7] Castellini F., Riccardi A., Lavagna M., and Bueskens C., "Launch vehicles multidisciplinary optimization, a step from conceptual to early preliminary design", International Astronautical Congress, Cape Town, October 2011.

[8] Deb K., Agrawal S., Pratap A., Meyarivan T., "A Fast Elitist Non-Dominated Sorting Genetic Algorithms for Multi-Objective Optimization: NSGA-II", KanGal report, Kanpur Genetic Algorithms Laboratories, 2000.

[9] Castellini F., Lavagna M.R. and Erb S., "Concurrent Optimization of Launcher Architectures and Ascent Trajectories with Global Optimization Algorithms", 58th International Astronautical Congress, Glasgow, UK, Sept. 2008.

[10] Dorigo M., Socha K., "Ant Colony Optimization for continuous domains", European Journal of Operational Research, Vol 185, 2008, pp. 1155, 1173.

[11] Kennedy J., Eberhart R., Particle Swarm Optimization, Neural Networks Proceedings., IEEE International Conference, 4 (1995), pp. 1942-1948.

[12] Nikolayzik T., Bueskens C., and Gerdts M., "Nonlinear large-scale Optimization with WORHP", AIAA/ATIO/ISSMO Multidisciplinary Analysis and Optimization Conference, Fort Worth, TX, September 2010

[13] www.worhp.de