RBF-metamodel driven multi-objective optimization and its applications

Tanja Clees, Nils Hornung, Igor Nikitin, Lialia Nikitina, Daniela Steffes-lai
Department of High Performance Analytics
Fraunhofer Institute for Algorithms and Scientific Computing
Sankt Augustin, Germany
Tanja.Clees|Nils.Hornung|Igor.Nikitin|Lialia.Nikitina|Daniela.Steffes-lai@scai.fraunhofer.de

***Abstract*—Metamodeling of simulation results with radial basis functions (RBF) is an efficient method for the continuous representation of objectives in parametric optimization. In the multi-objective case a detection of non-convex Pareto fronts is especially difficult, which is a point where many simple algorithms fail. In this paper we consider different formulations of the multi-objective optimization problem: as a sequential linear program (SLP), as a sequential quadratic program (SQP) and as a generic nonlinear program (NLP). We compare their efficiency and apply them in three realistic test cases. In the first application we consider a bi-objective optimization problem from non-invasive tumor therapy planning, where the typical goal is to maximize the level of tumor destruction and to minimize the influence to healthy organs. The second application case is safety assessment in automotive design. Here the crash intrusion in the driver and passenger compartment is minimized together with the total mass of the vehicle. The third application comes from the simulation of gas transport networks, where the goal is to fulfil the contract values, such as incoming pressure and outgoing flow delivery, providing the best energetic efficiency of the transport.**

Keywords-complex computing in application domains; advanced computing in simulation systems; advanced computing for statistics and optimization.

I. INTRODUCTION

This paper is a continuation of our previous work [1], significantly extended by new material. Particularly, in the paper [1] we have proposed three algorithms for the detection of non-convex Pareto fronts in multi-objective optimization problems, where the objectives are represented by RBF metamodels. The algorithms have been applied to a bi-objective optimization problem in focused ultrasonic therapy planning. In the current paper we give more details on the implementation of the algorithms and present two more applications. The first application is a bi-objective optimization problem in the field of automotive design. The statement of the problem has been given in [2], while the special methods for finding non-convex Pareto fronts described in this paper have never been applied to this problem and we do it here for the first time. The next application is a four-objective problem in optimization of gas transport networks, which has not been considered before. Here we give the detailed statement of the problem and find an optimal solution, providing 27% energy savings for the realistic network.

Related work on metamodeling techniques, multi-objective optimization and their industrial applications is presented in [2-8]. Our contribution with regard to the state-of-the-art is the development of efficient algorithms for detection of non-convex Pareto fronts in combination with RBF metamodeling of objectives.

RBF metamodeling is often used in applied problems for the continuous representation of optimization objectives from a discrete set of simulation results. It represents the interpolated function f(x) as a linear combination of special functions Φ() depending only on the distance to the sample points x_k:

$$f(x) = \sum_{k=1..Nexp} c_k \, \Phi(|x-x_k|). \qquad (1)$$

The coefficients c_k in (1) can be found from known function values in sample points $f(x_k)$ by solving a moderately sized linear system with a matrix $\Phi_{kn} = \Phi(|x_k-x_n|)$. A suitable choice for the RBF is the multi-quadric function $\Phi(r)=(b^2+r^2)^{1/2}$, b=Const, which provides non-degeneracy of the interpolation matrix for all finite datasets of distinct points and all dimensions [3]. RBF interpolation can be extended by adding polynomial terms, allowing the exact reconstruction of polynomial (including linear) dependencies and generally an improved precision of interpolation. The number of points should be greater than the number of available monomials to avoid overfitting. An adaptive sampling and a hierarchy of metamodels with appropriate transition rules are used for further precision improvement. RBF metamodels are directly applicable to the interpolation of high dimensional bulky data, e.g., complete simulation results can be interpolated at a rate linear in the size of the data, and even faster in combination with PCA-based dimensional reduction techniques. The precision can be controlled via a cross-validation procedure. RBF metamodels, enhanced in this way, are a part of our software tool for design parameter optimization DesParO [2,4,5].

Single objective optimization selects a point in parameter space, providing an optimum (e.g., maximum) of the objective function. In multi-objective optimization the optimum is not an isolated point but a hypersurface (Pareto front, [6]) composed of points satisfying a tradeoff property, i.e., none of the objectives can be improved without

simultaneous degradation of at least one other objective. E.g., for a bi-objective problem, the Pareto front is a curve on the 2D plane of the objectives bounding the region of possible solutions. Efficient methods have been previously developed for determining the Pareto front.

The simplest way is to convert multi-objective optimization to single objective one, by linearly combining all objectives into a single target function

$$t(x)= \sum w_i f_i(x) \qquad (2)$$

with user-defined constant weights w_i. Maximization of the target function (2) gives one point on the Pareto front, while varying the weights allows to cover the whole Pareto front. In this way only convex Pareto fronts can be detected, because non-convex Pareto fronts do not produce maxima but saddle points of the target function. For non-convex Pareto fronts this method just skips non-convex segments and jumps to the nearest convex part.

There are methods also applicable to problems with non-convex Pareto fronts. The non-dominated set algorithm (NDSA) finds a discrete analogue of the Pareto front in a finite set of points. For two points f and g in optimization criteria space the first one is said to be dominated by the second one if $f_i \le g_i$ holds for all i=1..Ncrit , where Ncrit is a number of objectives (optimization criteria). A point f belongs to the non-dominated set if there does not exist another point g dominating f. There is a recursive procedure [7] finding all non-dominated points in a given finite set. The drawback of the algorithm is an extremely large number of samples necessary to populate multidimensional regions for a good approximation of the Pareto front.

The normal boundary intersection method (NBI) [8] provides a good heuristic for sampling the Pareto front. The idea is to find individual minima of objectives, to construct their convex hull, to sample it, e.g., with Delaunay tessellation, to build normals in tesselation points and finally to intersect them with the boundary of the par → crit mapping. The approach has problems, e.g., at Ncrit>2, when non-Pareto points or not all Pareto points are covered, or if the number of minima is >Ncrit, when several local Pareto fronts can be mixed together.

Meanwhile, practical applications just require an elementary algorithm that performs a local improvement of a current design towards the optimum. Being iterated such an algorithm proceeds towards the Pareto front. For definiteness, an improvement direction in the space of objectives can be fixed, e.g., if at every step all objectives are improved by a given increment. The algorithm stops when no further improvement in the given direction is possible. Normally it happens when the solver reaches the Pareto front. Convex or non-convex Pareto fronts can be encountered and the algorithm should work equally efficient for both. The improvement can also stop at a non-Pareto boundary point. In this case the algorithm is allowed to return another point on Pareto front, which does not necessarily belong to the original improvement direction.

In Sections II-V we consider different approaches for this algorithm: sequential linear programming (SLP), sequential quadratic programming (SQP) and generic 1- or 2-phase nonlinear programming (NLP). We also consider the question of scalarization, i.e., the possibility to reformulate the multi-objective optimization problem as constrained optimization with a single objective, which allows to employ available NLP solvers for its solution. In Section VI we discuss the implementation details of the algorithms. In Sections VII-IX we describe the applications of the algorithms to the optimization problems in focused ultrasonic therapy planning, safety assessment in automotive design and simulation of gas transport networks, respectively. The relative benefits of the algorithms are summarized in Section X.

II. USING SEQUENTIAL LINEAR PROGRAMMING

Considering RBF metamodels (1) and linearizing the mapping y=f(x) using the Jacobi matrix $J_{ij}=\partial y_i/\partial x_j$, let us define a polyhedron of possible variations

$$\Pi_\varepsilon: \Delta y=J\Delta x \ , \ \Delta y \ge \varepsilon > 0 \ , \ -\delta \le \Delta x \le \delta, \qquad (3)$$
$$x_{min} \le x+\Delta x \le x_{max} \ , \ y_{min} \le y+\Delta y \le y_{max}.$$

Here we require that all criteria Δy are improved, parameter variations Δx are bounded in a trust region $[-\delta,\delta]$ for linear approximation, while parameters and criteria satisfy bounding box or other polyhedral restrictions in xy-space. By requiring in (3) that a maximally possible improvement of the criteria in Π_ε is achieved, we formulate a linear program that can be solved, e.g., by the simplex method [9] and repeated sequentially:

Algorithm SLP:

Solve LP: max ε, s.t. $(\Delta x,\Delta y)\in\Pi_\varepsilon$.

Repeat steps $x+\Delta x \rightarrow x$ until convergence.

The algorithm terminates at the Pareto front, where no further improvements are possible.

Property [9]: In general position the LP-optimum is achieved in the corners of the polyhedron Π_ε.

E.g., $\Delta y=\varepsilon$ corresponds to linear trajectories in y-space, $|\Delta x|=\delta$ corresponds to linear trajectories in x-space. Therefore, the method tends to generate linear trajectories in certain projections.

SLP is formulated above for the case dim(x)=dim(y). At dim(x)<dim(y) the multi-objective problem is ill defined, i.e., full dimensional regions in parameter space become Pareto equivalent. At dim(x)>dim(y) there are unstable directions from the kernel of Jacobi matrix Ker(J): $J\Delta x=0$, i.e., there are Δx not influencing Δy. These directions can be suppressed by the additional condition $J_\perp\Delta x=0$, where $J_\perp$ is the orthogonal complement to J, constructed, e.g., with the Gram-Schmidt algorithm.

Example: Let us consider a fold transform: $|y|=2|x|/(1+|x|^2)$, shown in Fig. 1 for the 2D case. An upper right arc corresponds to a global Pareto front (PF) max y_1,y_2. There is also a degenerate local PF at $y_{1,2}=-0$, corresponding to an image of $x_{1,2}=-\infty$.

The SLP algorithm generates trajectories shown by red lines in Fig. 1, in x-space in the left column and in y-space in the right column. The algorithm reconstructs correctly both global and local PFs, shown by blue points in the images. The bottom closeups demonstrate piecewise linear trajectories described above. Particularly, there is a dashed linear trajectory in y-space tending to the non-Pareto part of the boundary (nPF), which at a certain moment switches from the $\Delta y=\varepsilon$ corner to the $|\Delta x|=\delta$ corner, becomes curved and finally stops at the PF.

III. USING SEQUENTIAL QUADRATIC PROGRAMMING

The polyhedron Π_0 is defined as above (with $\varepsilon=0$). Let v be a fixed search direction in y-space, ε a constant. The following quadratic program [10] tries to perform $\Delta y=\varepsilon v$ steps if possible in Π_0:

Algorithm SQP:

Solve QP: min $||\Delta y-\varepsilon v||^2$, s.t. $(\Delta x,\Delta y)\in\Pi_0$

Repeat steps $x+\Delta x \rightarrow x$ until convergence.

Property [10]: In general position the QP-optimum can be achieved inside Π_0, in the corners of Π_0 or on the edges/faces of Π_0.

In the first case $\Delta y=\varepsilon v$ linear trajectories will be generated in y-space, in the second case $|\Delta x|=\delta$ linear trajectories will be generated in x-space, in the third case the trajectories become nonlinear.

Example: Fig. 1 also shows 5 trajectories from SQP method (green lines). Performance of SQP is analogous to SLP and we prefer to use SLP due to simplicity of its implementation.

IV. USING 1-PHASE NONLINEAR PROGRAMMING

Nonlinear target functions of the form $t(x)=\sum w_i\, crit_i^p$ can, under certain conditions, detect non-convex Pareto fronts. Here the target function is represented by a scaled L_p-norm with weights $w_i \geq 0$, $\sum w_i = 1$. Fig. 2 left shows the level curves for a 2D target function for different p. One has a straight line at p=1, a quadric at p=2, a superquadric at p>2 and a corner at p=∞.

Property (see Fig. 2 left): Nonlinear target functions can be used to detect non-convex PFs, if the curvature of the level curve exceeds the curvature of the PF.

Also at higher dimensions, considering the level set (LS) tangent to the PF, performing Taylor expansions of the LS and the PF: $z=u^TMu+o(u^2)$, where u,z are, respectively, the parallel and normal components to a common tangent hyperplane to the LS and the PF, and requiring $z_{LS}\geq z_{PF}$, one can reformulate the property above as positive definiteness for the difference of the curvature matrices M_{LS}-M_{PF}.

Note that L_∞ =max is applicable in any case (minmax method [11]), but the corresponding NLP will be non-smooth. Practically, one can use large finite p. It is also convenient to normalize y_i in [0,1] and take the logarithm of the target function for numerical stability. In this way one achieves a so called scalarization of the multi-objective optimization, i.e., the conversion of a multi-objective problem to a single objective one. As a result, the problem becomes solvable with standard NLP-solvers, e.g., Ipopt [12]. Here one can impose any additional constraints, e.g., require that $y(x) \leq c$. By putting $c=y_0$ one ensures that the result is better in all criteria than a starting point and finds only a corresponding segment of the PF. One can also leave $c=\infty$ and vary w_i to cover the whole PF.

Algorithm NLP1(c):

minimize $t(x)=\log \sum (w_i y_i)^p$, s.t. $y(x) \leq c$.

V. USING 2-PHASE NONLINEAR PROGRAMMING

The following algorithm combines the concepts of linear search from NBI and the optimization of a nonlinear target function. The first phase performs the linear search in a given direction v in y-space towards the PF and the second phase tries to perform further improvements (if possible). The problem is solvable with two calls to ipopt.

Algorithm NLP2:

NLP2.1: maximize t, s.t. $y(x)=y_0+tv$; result y_1;

NLP2.2: call NLP1(y_1); result y_2.

Properties (see Fig. 2 right):

if $y_1 \in$ PF, phase 2 quits immediately;

if $y_1 \in$ non PF boundary, trajectory is bounced to PF.

In NLP2.2 not the whole PF is targeted, but a smaller part ΔPF possessing better criteria values than y_1. Here one can use a smaller p, while even for too curved PFs the result y_2 will be still better than y_0 and y_1.

VI. IMPLEMENTATION DETAILS

The SLP and SQP algorithms with their specific definition of a polyhedron of variation and with their internal iteration loop are better to implement as separate program modules. The NLP algorithms can use existing NLP solvers, such as Ipopt, Snopt, Minos etc. An implementation can use one of the following interfaces:

AMPL environment (A Mathematical Programming Language, [13]) can be used to specify the optimization problem in a human-readable format. Fig. 3 top shows a snippet of code defining a constraint.

NL stub [14] is a standard format for optimization problems, containing the objective and constraints recorded in Polish prefix notation (PPN). The code forms an expression tree, whose branches are formed from a predefined set of operators (o…) extendable by user defined functions (f…), the leaves are numerical constants (n…) and variables (v…). Such a representation is suitable for the automatic differentiation of expressions using the chain rule. Fig. 3 middle shows the definition of a constraint in NL format.

Ipopt's C++ API [12] allows to inherit a new program from a base class TNLP (The Non-Linear Program or Template Non-Linear Program). The class possesses an evaluator of constraints and objectives, as well as their first and second derivatives, expressed in analytical or numerical form. Fig. 3 bottom represents the corresponding function prototypes.

Although AMPL and NL interfaces have a small overhead in comparison with the direct C++ API, in practice all these possibilities provide comparable performance.

VII. Application in focused ultrasonic therapy planning

Focused ultrasonic therapy is a non-invasive therapy using magnetic resonance tomography for the identification of tumor volume and focused ultrasound for the destruction of tumor cells. Numerical simulation becomes an important step for the therapy planning. Efficient methods for focused ultrasonic simulation have been presented in paper [15]. It uses a combination of the Rayleigh-Sommerfeld integral for near field and of the angular spectrum method for far field computations, which allows determining the pressure field in heterogeneous tissues. The bioheat transfer equation is used to determine the temperature increase in the therapy region. Thermal dose is defined according to the cumulative equivalent minutes metric (CEM, [16]) or the Arrhenius model [17] as a functional of temperature-time dependence in every spatial point in the therapy region. These methods have been accelerated by a GPU based parallelization and put in the basis of software FUSimlib (www.simfus.de), developed by our colleagues at the Fraunhofer Institute for Medical Image Computing.

3D visualization is used for the interpretation of the simulation results, in particular, for the detailed inspection of MRT images (magnetic resonance tomography), corresponding material model and spatial distribution of the resulting thermal dose, see Fig. 4. Stereoscopic 3D visualization in virtual environments based on modern 3D-capable beamers with DLP-Link technology (Digital Light Processing), described in more details in [18], is especially suitable for this purpose. Such commonly available beamers do not require special projection screens and can turn every regular office to a virtual laboratory providing full immersion into the model space. We use 3D visualization software Avango (www.avango.org), an object-oriented programming framework for building applications of virtual environments. Our interactive application overlays three voxel models: The original MRT sequence, the material segmentation and the resulting thermal dose. The user can mix the voxel models together, interactively changing their levels of transparency, set the breathing phase, cut the model with a clipping plane, etc.

TABLE I. Bi-objective optimization in focused ultrasonic therapy planning, problem characteristics

Parameter bounds:	**Timing per solution**
frequency 0.25…0.75 MHz	@ 3GHz Intel i7:
ini.speed 0.23...0.282 m/s	
Criteria bounds:	SLP 7ms
∑TDin 0…3000 eq.min	NLP1 16ms
∑ TDout 0…6000 eq.min	NLP2 13ms+12ms

A generic workflow for ultrasonic therapy simulation has been described in our paper [19]. Numerical simulation with FUSimlib software uses a 512 x 512 x 256 voxel grid. Ultrasound has been focused in the center of the target zone for the neutral breath state. The result after 10 seconds of exposure time (200 steps x 0.05sec) has the form of spatial distributions of the pressure amplitude, the temperature and the thermal dose. Fig. 4 top-right shows a typical result for thermal dose on slice 97/256 near the focal point. The frequency of the transducer is taken as an optimization parameter controlling the focused ultrasonic therapy simulation. The other one, initial particle speed, is proportional to an acoustic intensity emitted by the transducer [15]. The objective of therapy planning is a maximization of the thermal dose inside the target zone (TDin) and a minimization of the thermal dose outside (TDout). The thermal doses are defined as sums of the thermal dose over corresponding voxels, ∑TDin / ∑TDout. The variation range of the optimization parameters was regularly sampled with 25 simulations, from which 16 fall in the region of interest, shown in Fig. 5 by red points. Our RBF metamodel constructed on simulation results is used to oversample the region by green points, from which the discrete method NDSA selects the Pareto front, shown by blue points. We see that the Pareto front is of non-convex type. Magenta lines show the application of the three continuous methods described above. The trajectories generated by SLP and NLP2 coincide in every detail. Even bouncing from the non-PF boundary works similarly, although the mechanisms of this bouncing are different. NLP1 with p=8 and w_1=0.01, 0.15, 0.27, 0.5, 0.99 produces the other set of trajectories. Table I shows a summary of the problem characteristics. SLP provides the best performance for the given application case. On the other hand, NLP

provides an easier integration with existing scalar solvers. In the NLP class, NLP1 is faster than NLP2 for bounced trajectories, otherwise NLP2 is faster. Numerically NLP2 (with small p) is less singular than NLP1 (with large p) and, therefore, it is more robust for the detection of strongly curved Pareto fronts.

VIII. Application in Audi B-pillar Crash Test

We apply the algorithm NLP2 for the detection of Pareto fronts in automotive crash test simulation, described in [2]. The PamCrash software (www.esi.com.au/Software/PAM-CRASH.html) is used for simulation. The model of a B-pillar shown in Fig. 7 contains 10 thousand nodes. 45 timesteps of crash process are simulated. 101 simulations were made by varying two design parameters, i.e., the thicknesses of two layers composing the B-pillar. The optimization objective is the simultaneous minimization of the total mass of the B-pillar and of the crash intrusion in the contact area. Crash intrusions in the driver and passenger compartment are commonly considered as critical safety characteristics of car design, while the total mass influences other important characteristics, i.e., fuel consumption, CO2 emissions and production costs. Two extreme designs corresponding to lower and upper bounds of both thicknesses are shown in Fig. 7.

Further, we apply RBF metamodeling to represent the relation between design parameters and optimization criteria and study this problem in our interactive optimization tool DesParO, see Fig. 6. At first, we impose constraints on the optimization criteria, as indicated by red ovals in Fig. 6 top. As a result, islands of available solutions become visible along the axes of design parameters. The islands are combined cross-like, as shown in Fig. 6 bottom. For these combinations both constraints on mass and intrusion are satisfied, while all alternative combinations violate the constraints. In this way a complex structure of the Pareto front in the considered problem is revealed.

Fig. 8 shows the results of the NDSA and NLP2 methods on two different projections. On the left the Pareto front on the criteria plane is presented. Blue points indicate the first piece of the Pareto front found by NDSA and corresponding to the first island of solutions. Meanwhile NLP2 trajectories sometimes stop earlier and indicate the second piece of the Pareto front shown by a dashed line. Fig. 8 right part shows a different projection, where both pieces become visible. There is a hill separating solutions like a watershed, so NLP2 trajectories go to the one or to the other side dependently on a starting point. This projection produces also a fold, a small overlap near the second piece of the Pareto front. It does not disturb the convergence of the NLP2 algorithm, the trajectory jumps directly from the starting point to the second piece of the Pareto front, displayed by the dashed line in the figure.

We note that NDSA considers a dataset as a cloud of discrete points in the space of criteria, without any notion of continuity in parameter and criteria space, and detects only one piece of the Pareto front (global optimum). NLP2 is a continuous method and detects also the second piece (local optimum). In the considered problem both optima are close to each other and represent the underlying symmetry of the problem. Indeed, the Pareto optimal solution for this problem belongs to a boundary of the parameter space. It corresponds to a minimal thickness of one layer and varied thickness of the other layer, representing a compromise between stiffness and mass. Two possibilities in a choice of the minimal layer become two pieces of the Pareto front.

TABLE II. Bi-objective optimization in Audi B-pillar crash test simulation, problem characteristics

Parameter bounds:	
thickness1,2	0.5…3 mm
Criteria bounds:	
intrusion	0…174 mm
mass	7.8…26.9 kg

IX. Application in gas transport networks

The simulation of gas transport networks is performed by a software package Mynts (Multi-phYsics NeTwork Simulator, www.scai.fraunhofer.de/en/business-research-areas/high-performance-analytics-en/products/mynts.html) developed in our group. A small training network used here for experiments is shown in Fig. 9. It contains 100 nodes, 111 pipes and other connecting elements. We note that real life problems are much larger. In cooperation with our partners we solve stationary and transient problems for gas networks with ten thousands of elements.

The network of Fig. 9 has two supply nodes with specified pressure values (PSETs) and three consumer nodes with specified flow values (QSETs). There are two compressor stations, providing the necessary throughput in the network, see Fig. 10. Every station consists of two separate compressors, each can be configured to provide a fixed output pressure (SPO) or a fixed flow value (SM). In the considered scenario one compressor is set to SPO mode and three others to SM mode and these four values are used as optimization parameters. The purpose of the optimization is to run the compressors at minimal possible power (POW) sufficient to satisfy all contract values, such as PSETs and QSETs. The result is a particular solution of a network feasibility problem, possessing the best energetic efficiency.

For this study we have prepared 1000 simulations in the bounds given in Table III. Fig. 11 shows a solution of the optimization problem in our DesParO Metamodel Explorer. It is achieved at a minimal SPO value for the first

compressor and a particular distribution of SM values, corresponding to individual properties of the three other compressors. The result of the NLP2 optimization is given in Table IV. We see that the optimization provides 27% energy savings relative to the starting point.

TABLE III. FOUR-OBJECTIVE OPTIMIZATION IN GAS TRANSPORT NETWORK SIMULATION, PROBLEM CHARACTERISTICS

Parameter bounds: SPO1 71…91 bar SM2-4 400...600 x1000m^3/h
Criteria bounds: POW1-4 2…16 MW

TABLE IV. FOUR-OBJECTIVE OPTIMIZATION IN GAS TRANSPORT NETWORK SIMULATION, STARTING POINT AND OPTIMAL SOLUTION

	STARTING POINT	OPTIMAL SOLUTION
SPO1	81	71
SM2	500	510
SM3	500	508
SM4	500	502
POW1	8.5	6.3
POW2	8.6	6.4
POW3	8.4	6.2
POW4	7.8	5.6

X. CONCLUSION

Several algorithms of continuous multi-objective optimization applicable to the detection of non-convex Pareto fronts have been discussed: sequential linear programming (SLP), sequential quadratic programming (SQP) and generic 1- or 2-phase nonlinear programming (NLP1,2). Performance of SQP is analogous to SLP and we prefer to use SLP due to simplicity of its implementation. Scalarization, i.e., the reformulation of the multi-objective optimization problem as constrained optimization with a single objective, allows to employ available NLP solvers for its solution. The algorithms have been applied to a realistic test case in focused ultrasonic therapy planning. In the given problem SLP possesses the best performance, while NLP provides an easier integration with existing scalar solvers. NLP1 is faster than NLP2 for bounced trajectories, otherwise NLP2 is faster. Numerically NLP2 is less singular than NLP1 and is therefore more robust for the detection of strongly curved Pareto fronts. All these optimization methods provide real-time performance necessary for the interactive planning of focused ultrasonic therapy. Further, NLP2 has been applied to a multi-objective optimization problem in the safety assessment of automotive design. Here the Pareto front is also non-convex and consists of two separate pieces, global and local parts of the Pareto front. NDSA detects only the global part, while NLP2 finds both. Finally, we have applied NLP2 to improve the energetic efficiency of a gas transport network, where the optimization allows to achieve 27% energy savings.

REFERENCES

[1] T. Clees et al., “RBF-metamodel Driven Multi-objective Optimization and its Application in Focused Ultrasonic Therapy Planning”, in C.-P. Rückemann et al. (Eds.), ADVCOMP 2015, The Ninth International Conference on Advanced Engineering Computing and Applications in Sciences, July 19-24, 2015, Nice, France, pp. 71-76.

[2] T. Clees et al., “Analysis of bulky crash simulation results: deterministic and stochastic aspects”, in N.Pina et al. (Eds.): Simulation and Modeling Methodologies, Technologies and Applications, AISC 197, Springer 2012, pp. 225-237.

[3] M. D. Buhmann, “Radial Basis Functions: theory and implementations”, Cambridge University Press, 2003.

[4] G. van Bühren et al., “Aspects of adaptive hierarchical RBF metamodels for optimization”, Journal of computational methods in sciences and engineering JCMSE 12 (2012), Nr.1-2, pp. 5-23.

[5] T. Clees et al., “Nonlinear metamodeling of bulky data and applications in automotive design”, in M. Günther et al. (eds), Progress in industrial mathematics at ECMI 2010, Mathematics in Industry (17), Springer, 2012, pp. 295-301.

[6] M. Ehrgott and X. Gandibleux (Eds.), “Multiple criteria optimization: state of the art annotated bibliographic surveys”, Kluwer 2002.

[7] H. T. Kung et al., “On finding the maxima of a set of vectors”, Journal of the ACM, 22(4), 1975, pp. 469-476.

[8] G. Eichfelder, “Parametergesteuerte Lösung nichtlinearer multikriterieller Optimierungsprobleme”, Friedrich–Alexander-Universität Erlangen–Nürnberg, Dissertation 2006.

[9] G. Dantzig, “Linear programming and extensions”, Princeton University Press and the RAND Corporation, 1963.

[10] R. Fletcher, “Practical Methods of Optimization”, Wiley 2000.

[11] D. Müller-Gritschneder, “Deterministic Performance Space Exploration of Analog Integrated Circuits considering Process Variations and Operating Conditions”, Technische Universität München, Dissertation 2009.

[12] A. Wächter, “Short Tutorial: Getting Started With Ipopt in 90 Minutes”, IBM Research Report, 2009.

[13] R. Fourer, D. M. Gay, and B. W. Kernighan, “AMPL: A Modeling Language for Mathematical Programming”, Cengage Learning, 2002.

[14] D. M. Gay, “Writing .nl Files”, Albuquerque, Sandia National Laboratories, Technical report, 2005.

[15] J. Georgii et al., “Focused Ultrasound - Efficient GPU Simulation Methods for Therapy Planning”, in Proc. Workshop on Virtual Reality Interaction and Physical Simulation VRIPHYS, Lyon, France, 2011, J. Bender, K. Erleben, and E. Galin (Editors), Eurographics Association 2011, pp. 119-128.

[16] S. Nandlall et al., “On the Applicability of the Thermal Dose Cumulative Equivalent Minutes Metric to the Denaturation of Bovine Serum Albumin in a Polyacrylamide Tissue Phantom”, in Proc. 8th Int. Symp. Therapeutic Ultrasound (AIP), 1113, 2009, pp. 205-209.

[17] J. A. Pearce, “Relationships between Arrhenius models of thermal dose damage and the CEM 43 thermal dose”, in T. P. Ryan (ed.), Proc. of Energy-based Treatment of Tissue and Assessment V, SPIE, 2009, p. 7181.

[18] T. Clees et al., “Focused ultrasonic therapy planning: Metamodeling, optimization, visualization”, J. Comp. Sci. 5 (6), Elsevier 2014, pp. 891-897.

[19] T. Clees et al., “Multi-objective Optimization and Stochastic Analysis in Focused Ultrasonic Therapy Simulation”, in Proc. of SIMULTECH 2013, SCITEPRESS, 2013, pp. 43-48.

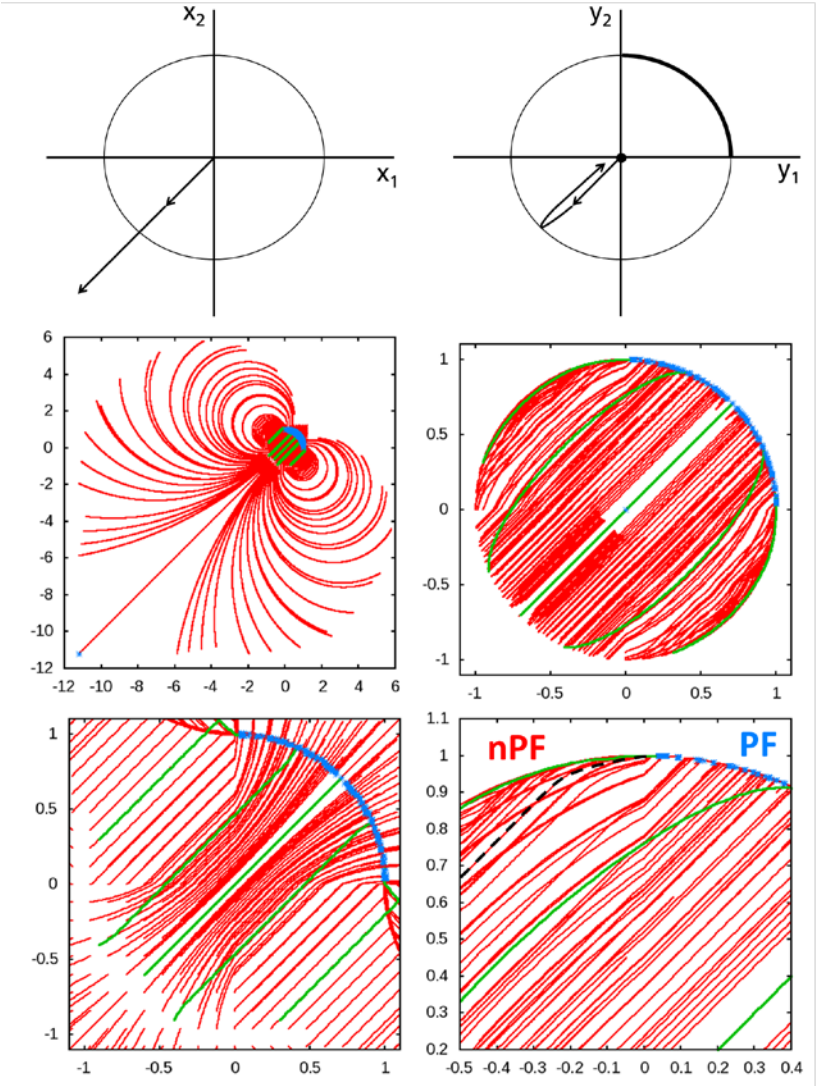

Figure 1. Pareto front detection for 2D fold transform (see Sections II-III for details).

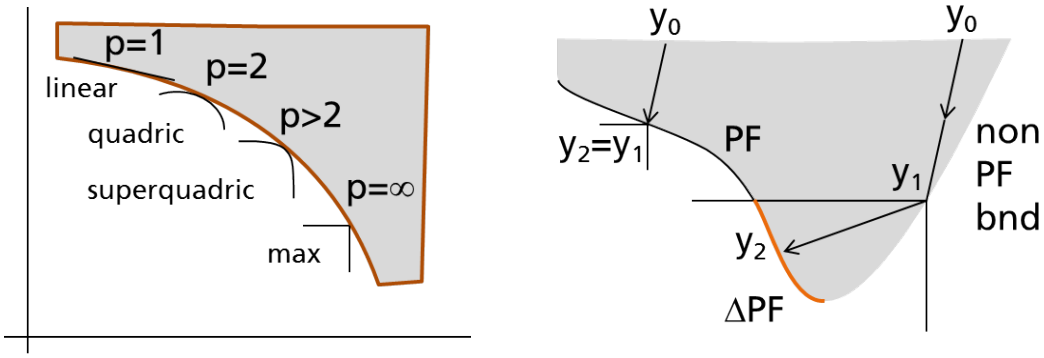

Figure 2. Scalarization of multi-objective optimization problem. On the left: algorithm NLP1; on the right: algorithm NLP2.

```
maximize obj: t*1e6;
subject to y0constr:
y0=sqrt(0.01+((x0-250000)/500000-0)**2+((x1-0.23)/0.052-0)**2)*(4208.98)+sqrt(0.01+((x0-250000)/500000-
0)**2+((x1-0.23)/0.052-0.25)**2)*(-938.255)+sqrt(0.01+((x0-250000)/500000-0)**2+((x1-0.23)/0.052-
0.5)**2)*(-310.969)+sqrt(0.01+((x0-250000)/500000-0)**2+((x1-0.23)/0.052-0.75)**2)*(-
1405.99)+sqrt(0.01+((x0-250000)/500000-0)**2+((x1-0.23)/0.052-1)**2)*(-527.836)+sqrt(0.01+((x0-
250000)/500000-0.25)**2+...
```

```
C0 o54 28 o2 n-4208.98 o39 o54 3 n0.01 o5 o3 o0 n-250000 v0 n5e+05 n2 o5 o3 o0 n-0.23 v1 n0.052 n2 o2
n938.255 o39 o54 3 n0.01 o5 o3 o0 n-250000 v0 n5e+05 n2 o5 o0 n-0.25 o3 o0 n-0.23 v1 n0.052 n2 o2
n310.969 o39 o54 3 n0.01 o5 o3 o0 n-250000 v0 n5e+05 n2 o5 o0 n-0.5 o3 o0 n-0.23 v1 n0.052 n2 o2
n1405.99 o39 o54 3 n0.01 o5 o3 o0 n-250000 v0 n5e+05 n2 o5 o0 n-0.75 o3 o0 n-0.23 v1 n0.052 n2 o2
n527.836 o39 o54 3 n0.01 o5 o3 o0 n-250000 v0 n5e+05 n2 o5 o0 n-1 o3 o0 n-0.23 v1 n0.052 n2 o2 ...
```

```
bool eval_f(Index n, const Number* x, bool, Number& f); // objective function
bool eval_grad_f(Index n, const Number* x, bool, Number* gf); // gradient of objective function
bool eval_g(Index n, const Number* x, bool, Index m, Number* g); // constraints
bool eval_jac_g(Index n, const Number* x, bool, Index m, // Jacobian of constraints
                      Index njac, Index* iRow, Index *jCol, Number* values);
bool eval_h(Index n, const Number* x, bool, Number obj_factor, Index m, const Number* lambda, bool,
                      Index nhess, Index* iRow, Index* jCol, Number* values); // Hessian of Lagrangian
```

Figure 3. Example of NLP code in different interfaces. Top: AMPL model file. Center: NL stub file. Bottom: Ipopt C++ API.

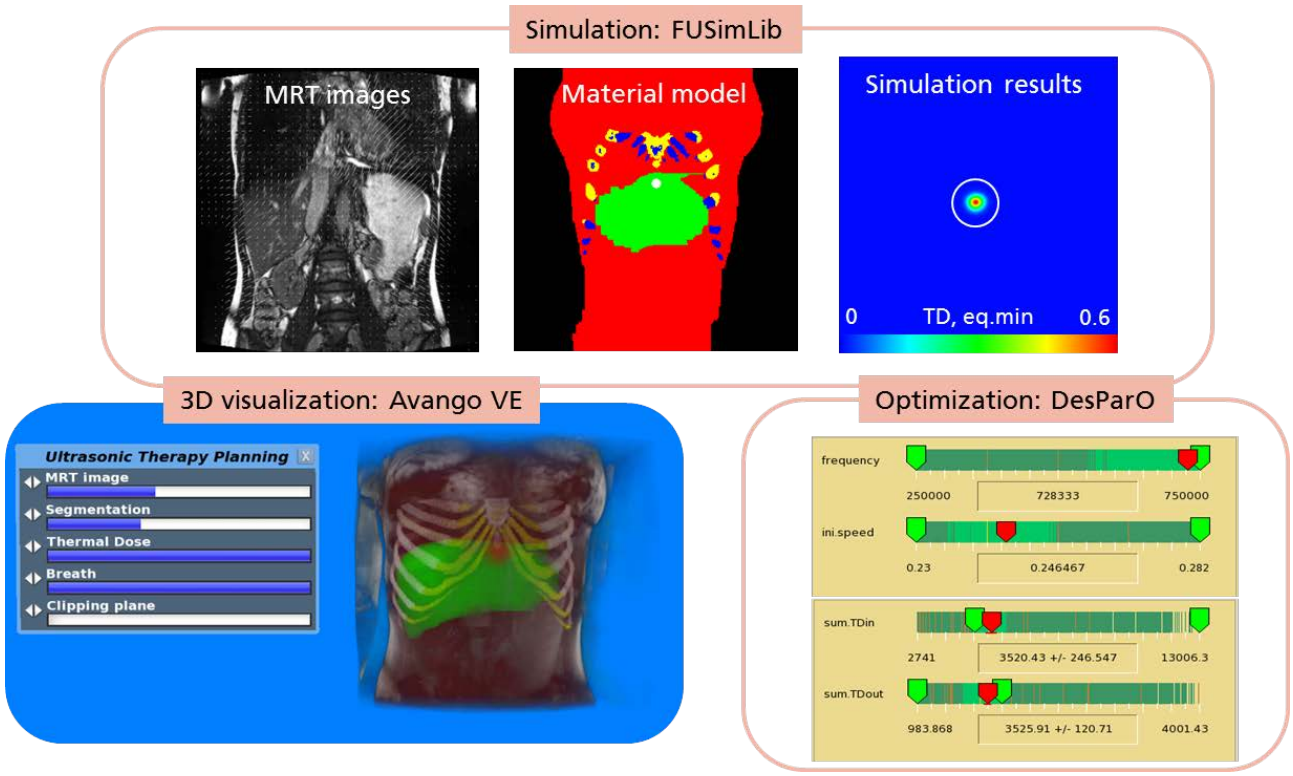

Figure 4. Focused ultrasonic therapy planning and its software components.

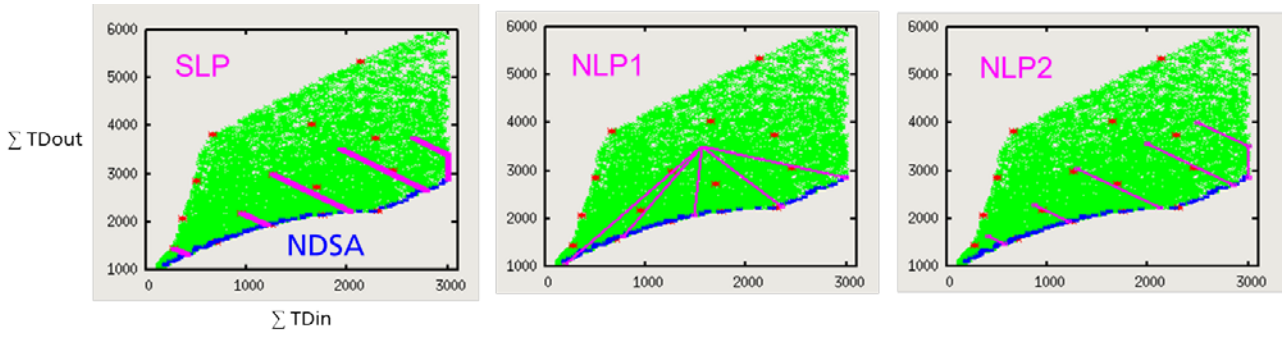

Figure 5. Non-convex Pareto front in focused ultrasonic therapy planning, comparison of different methods.

Figure 6. Studying Audi B-Pillar crash test in DesParO Metamodel Explorer. Top: an attempt to minimize simultaneously intrusion and mass by setting upper constraints on these criteria. Bottom: two islands of available solutions become visible.

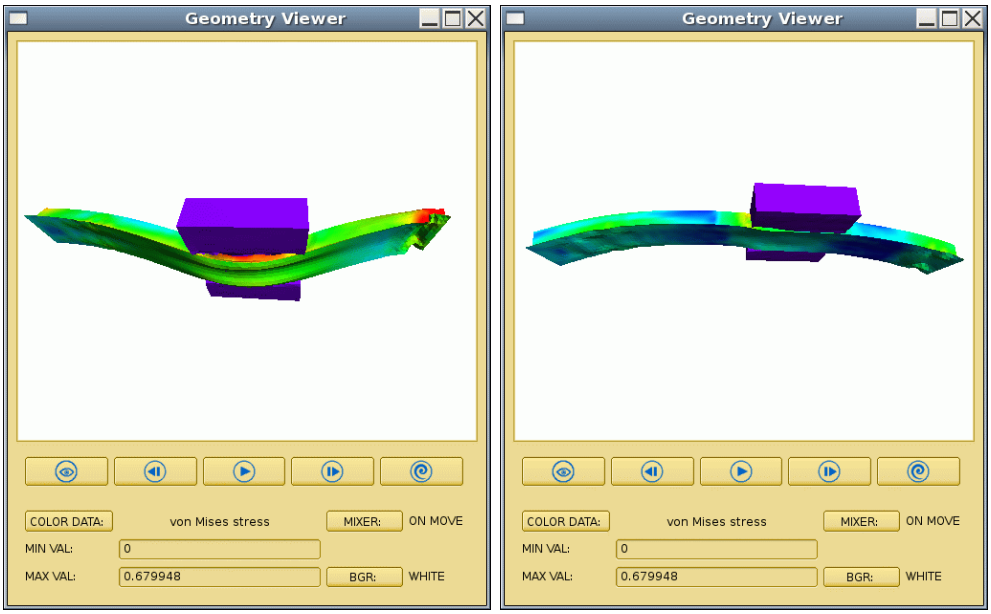

Figure 7. Audi B-Pillar crash test in DesParO Geometry Viewer. Two extreme designs are shown. On the left: small stiffness, small mass. On the right: large stiffness, large mass.

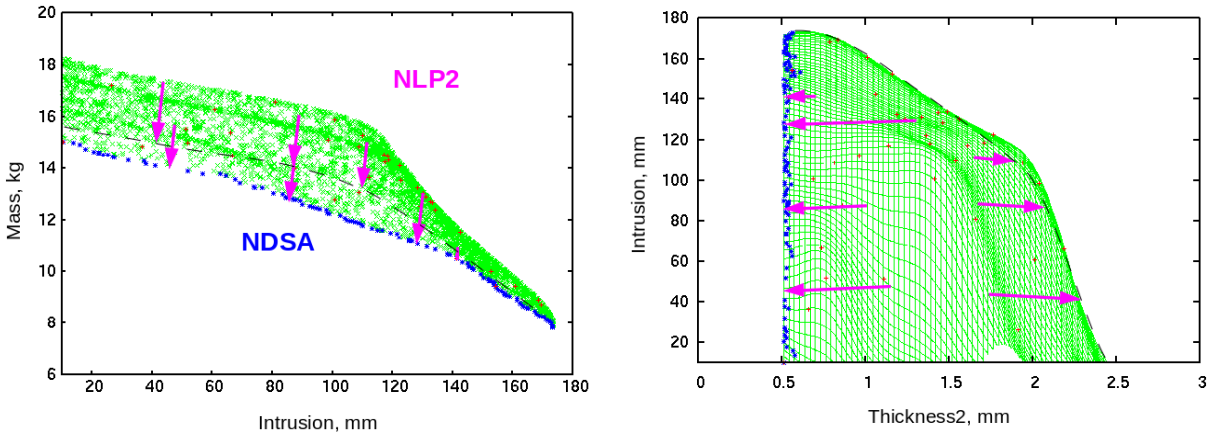

Figure 8. Pareto front of Audi B-Pillar crash test. On the left: intrusion vs mass projection. On the right: intrusion vs thickness2 projection. Red points show simulation results, green area - RBF interpolation, blue points - global Pareto front found with NDSA. Magenta arrows are trajectories of NLP2 algorithm. Dashed line shows a local Pareto front, corresponding to the second island of solutions.

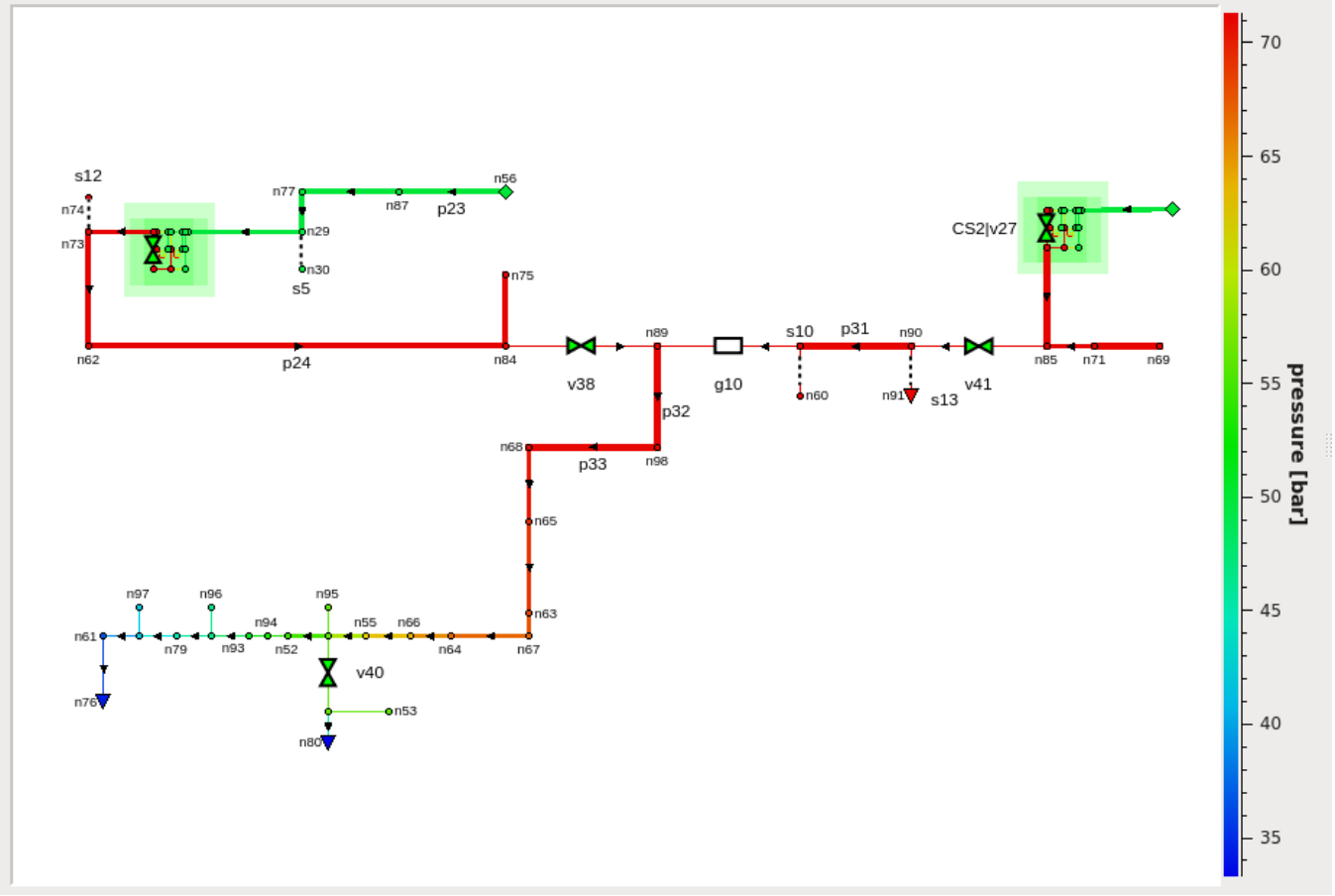

Figure 9. Gas transport network simulation in Mynts, the network topology with the resulting pressure distribution, shown by color.

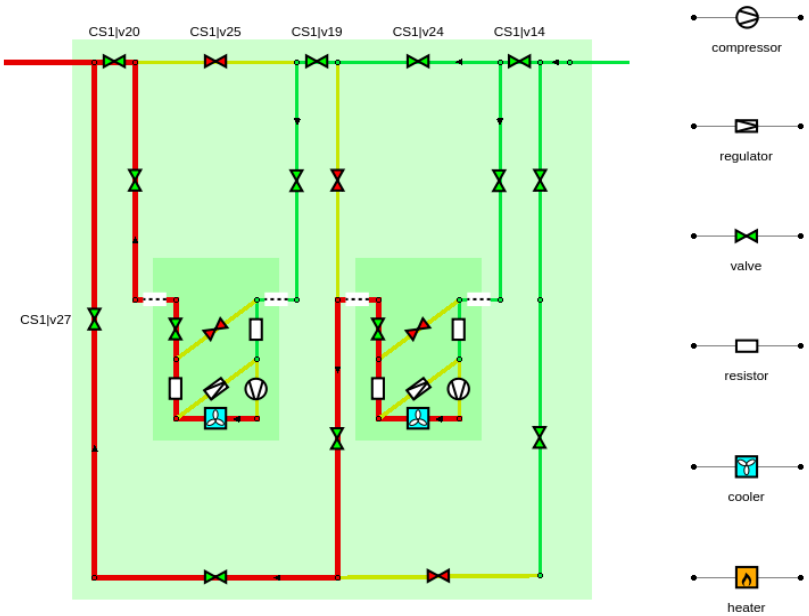

Figure 10. Gas transport network simulation in Mynts, closeup to a compressor station.

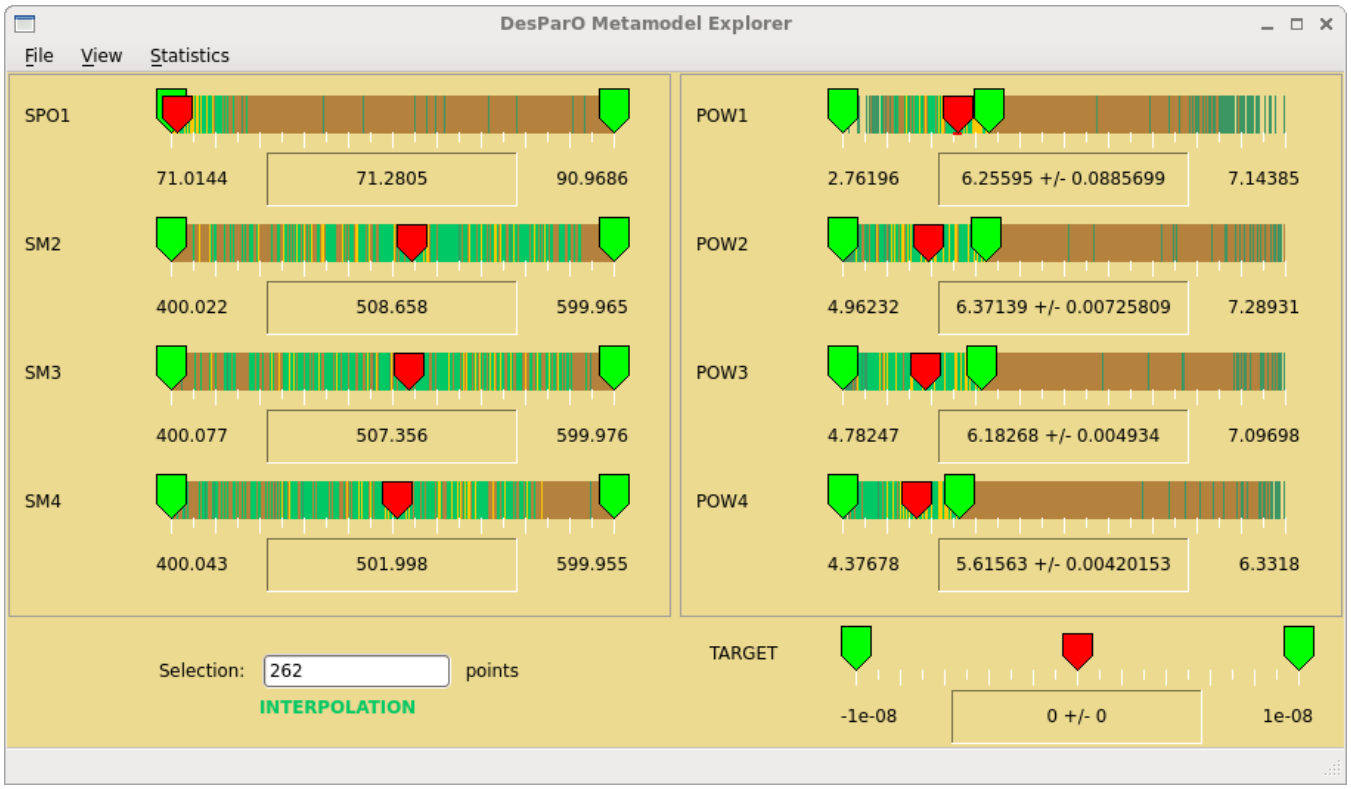

Figure 11. Four-objective optimization problem for gas transport network simulation in DesParO Metamodel Explorer.