Proceedings of the
44th IEEE Conference on Decision and Control, and
the European Control Conference 2005
Seville, Spain, December 12-15, 2005

ThB05.3

Tuning of a PID controller Using a Multi-objective Optimization Technique Applied to A Neutralization Plant

Andrey Popov, Adel Farag and Herbert Werner
Hamburg University of Technology, Institute of Control Engineering
andrey.popov@gmx.net, adel_farag@ieee.org, h.werner@tu-harburg.de

Abstract— Most control engineering problems are characterized by several, often contradicting, objectives, which have to be satisfied simultaneously. Two widely used methods for finding the optimal solution to such problems are aggregating to a single criterion, and using Pareto-optimal solutions. Here we propose a Genetic Algorithm (GA) approach using a combination of both methods to find a fixed-gain, discrete-time PID controller for a chemical neutralization plant. Known to be highly non-linear and with varying time delay, this plant provides a challenging testbed for nonlinear control strategies. Experimental results confirm that a multi-objective, Pareto-based GA search gives a better performance than a single objective GA. The former method was also used to design a gain-scheduled PID controller, for which also experimental results are shown.

I. INTRODUCTION

Most control engineering problems are characterized by several objectives that must be satisfied. Engineers are often faced with design problems, where a controller is needed that provides e.g. a fast response, small overshoot, no oscillation and economical control. There are mainly two ways of tackling this problem: (i) aggregating the objectives to a single objective or (ii) solving a multi-objective optimization problem (Pareto-based method). Aggregating several objectives into a single objective has the advantage of solving a simpler problem, but on the other hand many design iterations are required to obtain an acceptable compromise. On the other hand, the multi-objective approach is claimed to lead to a set of solutions each of which dominates the others in some sense.

This paper compares these two techniques by applying them to a practical controller design problem: to design a gain-scheduled PID controller for a chemical titration and neutralization plant (TINA). This plant is known to be highly nonlinear, which makes it suitable for comparison. In [1], [2] and [4] different solutions to this problem are proposed. Here, however we consider the design of a standard PID controller. This controller type is still the one most widely used in industry [5]. The ease of manual fine-tuning of the three coefficients makes PID controllers the preferred choice for many processes. Here we consider a disturbance rejection problem and the design of both a fixed-coefficient and a gain-scheduled discrete-time PID controller.

Although a variety of PID tuning techniques is available [5], the most efficient way to tune the parameters of a PID controller for non-linear or high-order systems is direct optimization. This requires however the solution of non-convex problems. Genetic algorithms (GA) are well suited for a variety of optimization problems in the control field [3]. The single-objective approach is implemented using the GA and direct search toolbox for Matlab 7.0. The multi-objective algorithm used here is the Strength Pareto Evolutionary Algorithm (SPEA2) [8], [7], which is one of the most powerful multi-objective techniques, and has outperformed many other algorithms (e.g. the one proposed in [6]).

II. TITRATION AND NEUTRALIZATION PLANT

A laboratory scale chemical plant, allowing titration and neutralization of chemical substances, is available at the TUHH Institute of Control Engineering and was used to neutralize hydrochloric acid (HCl) and Natrium Hydroxide ($NaOH$). Fig. 1 shows a diagram of the reaction tank. The acid inflow is controlled by a pump, and the base inflow by a valve. The pH value of the acid is 1.7 ± 0.05 and the pH of the base is 12.5 ± 0.1. Separate loops are controlling the liquid level and the temperature in the reaction tank to keep it at 210 ± 10 mm and $21 \pm 1\,^\circ$C, respectively. Those low-level loops and the mixer in the tank operate independently and are not considered here.
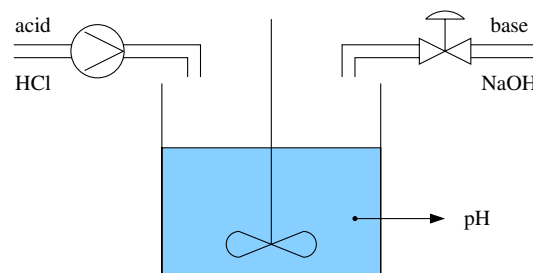

Fig. 1.   Reaction tank

The neutralization plant is characterized by strongly non-linear behaviour, having operating points with very high gain and operating points with very low gain, as well as saturation and time variations caused by buffering species (salt, contaminations, etc). An additional difficulty is due to the significant longer time delay of the base actuator (valve) than that of the acid actuator (pump). Two titration curves (one showing transition from acid to base and one the opposite direction) are shown in Fig. 2.

Depending on its sign, the control signal is applied either to the valve (positive values of the control signal) or the
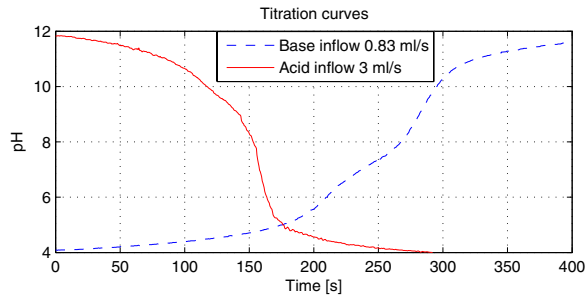
Fig. 2. Titration curves

pump. Both have non-linear characteristics, shown in Fig. 3. As a first step, the dead-zone of the actuators was compensated for by adding an offset, and the characteristics were linearized around 0 by increasing the gain for the base signal.
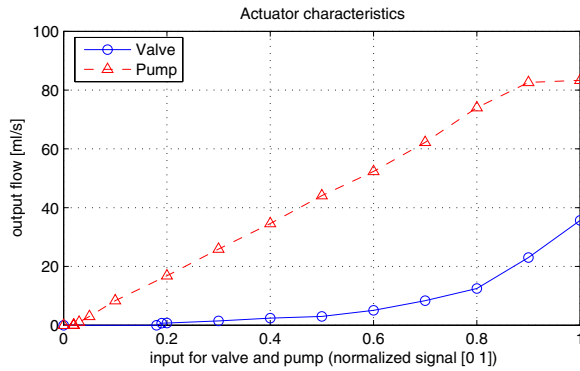

Fig. 3. Actuator characteristics

III. PROBLEM FORMULATION

A stabilization and disturbance rejection task is considered. The pH value in the reaction tank should be kept constant at 7 and disturbances applied to the input of the plant should be rejected.

Experimental data from the plant running in closed-loop operation with varying disturbance signals (Fig. 4) have been gathered. During the experiment a proportional controller was used to keep the pH value around 7 and avoid saturation. The same disturbance signal was used again later in the tuning process. The data set was then used to train a neural network (NN). Using the NN identification toolbox ([9]), a satisfactory model of the plant was obtained. Comparison of the model output with a cross-validation data set is shown in Fig. 5. One can clearly see that step disturbances of same amplitude but opposite directions (acid and base) have different effects. For large disturbance amplitudes the compensation of acid takes about twice the time as that for base.

IV. FIXED GAIN PID CONTROL

Two design objectives are considered: (1) minimal error (e) between the reference signal and the output and (2) minimal control (u). Whereas minimizing the first objective will provide good reference tracking and better disturbance
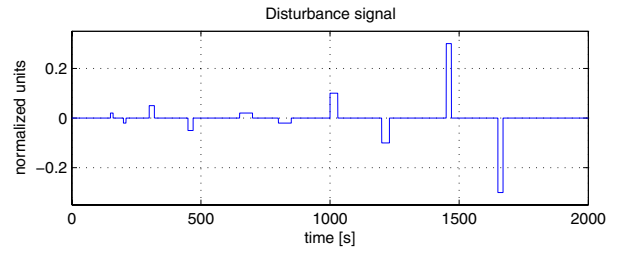

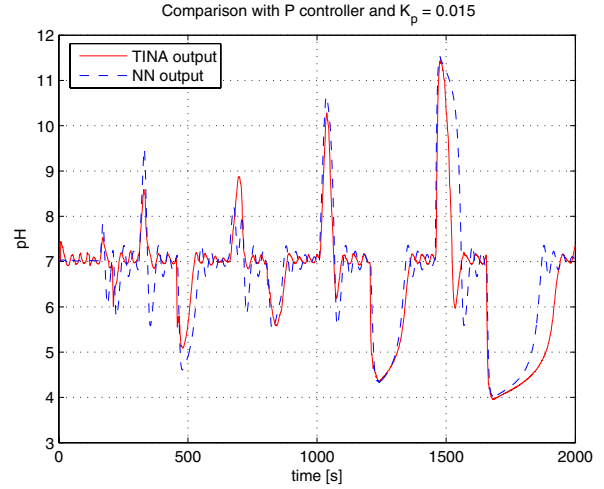Fig. 4. Applied disturbance signal

Fig. 5. Model validation

rejection, minimizing the second reduces the quantity of of acid and base and thus the cost of the control. The mathematical description of these objectives are given below

$$J_1 = \int_0^{T_s} e^2(t)dt \tag{1}$$

$$J_2 = \int_0^{T_s} |u(t)|dt \tag{2}$$

where $T_s = 2000s$ is the simulation time.

The first objective is the integral squared error, so large amplitudes are penalized. On the other hand, J_2 is a good measure of the total acid and base quantities required for control. Using the NN non-linear model and J_2 as the integral of the absolute error makes the optimization problem non-convex. Here we will use only these two objectives, the Pareto-surface of which can be easily displayed. However, more objectives (e.g. separate measures for acid and base) can be introduced if necessary.

First we use the objective aggregation method and combine the two objectives into a single one. Then we compare the results with those obtained by multi-objective optimization. In both cases we have three design parameters: K_P, K_I and K_D of the PID controller. The sampling time is $T_s = 0.5s$. The transfer function of the PID controller is

$$K(z) = K_P + K_I \frac{KT_s}{z-1} + K_D \frac{2z-2}{(T_s+2T_c)z+T_s-2T_c}$$

where $T_c = 0.5ms$ is the time constant for the derivative.

A. Objective Aggregation

When using objective aggregation, weight factors are introduced that give different priority to different objectives. Here we need only one weight W to aggregate J_1 and J_2 to

$$J_a = J_1 + WJ_2 \qquad (3)$$

Before starting the optimization, one needs to specify a value for W. At this stage factors like the "raw" values of the objective functions and their importance could be considered when choosing the appropriate weight. When, however, such information is not available, different values of W are used and the corresponding system behaviors are compared.

A population size of 20 individuals for the GA is chosen. Fragment-exponential encoding of the PID coefficients is used, by which each coefficient is encoded by two design variables using the form $K = p_1 10^{p_2}$. This allows the GA to search a wide range of values, with a small change of the exponential variable. Preliminary optimization runs show that 200 generations are sufficient from the point of view of convergence and repeatability of the results.

Simulations for 52 different weights were performed. With the obtained PID coefficients the separate values of J_1 and J_2 were computed and the results are shown in Fig. 6. Seven dominated points, which are above and to the right of the plot area, are not shown. At first a logarithmic scale for the weight was used from 10^{-1} to 10^2 and 7 points were found. It is important to note that the computation time for each weight value is approximately 45 min on a 2.6 GHz Pentium IV processor. The results were used to define the precise search area: from 0.1 with step size 0.1 to 1; from 1 with step size 1 to 10 and from there with step size 10 up to 100. The resulting points on the Pareto surface were still clustered in two areas, and the edge elements were found to be 50 and 70. A step size of 1 was used to scan the area between them, but the corresponding results were still on either side of the gap. Significantly increasing the population size (500) did not help finding points in the gap. Due to the long computation time this approach was not pursued further.

The main reason for the gap existing in Fig. 6 is the high sensitivity of the aggregated objective function ($J_1 + WJ_2$) at some regions of the Pareto-set to changes in the tuning parameter W. For example a very small change of the weight between 56 and 61 gives a huge change of the trade-off between J_1 and J_2. The elements in-between were converging either to one of the sides or to a dominated point far from the surface.

B. Multi-Objective GA (MOGA)

In contrast to the single-objective algorithms, which try to find a single solution of the problem, the multi-objective technique searches for the optimal Pareto set directly. A solution is said to be Pareto-optimal (non-dominated), if there exists no other solution which simultaneously improves the values of both objectives (J_1 and J_2). GA are particulary well suited for multi-objective techniques, because they work with a population of solutions rather than a single solution.
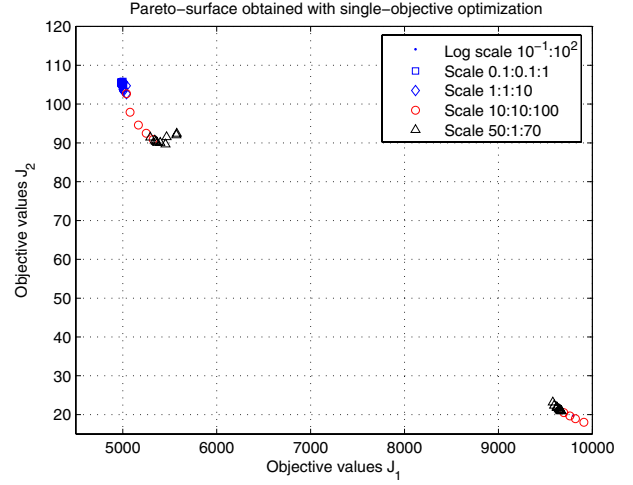

Fig. 6. Single-objective optimization results

The algorithm selected here is SPEA [7], [8]. The available C code, PISA format ([13]) version of the algorithm, was modified to allow system simulation and calculation of the objective functions in a MATLAB environment.

Here again fragment-exponential encoding of the PID coefficients was used. Population size and Pareto-set size of 50 points were found to give a good distribution over the Pareto-surface. At the same time 200 generations are enough for the algorithm to converge to a set close to the expected optimal one. The resulting Pareto-surface is shown in Fig. 7.
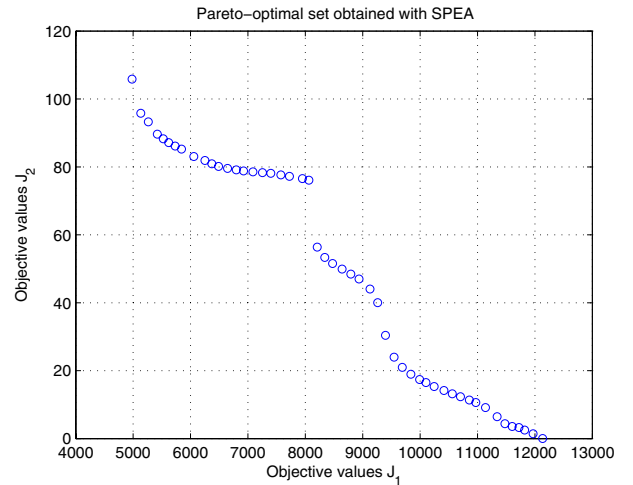

Fig. 7. Multi-objective optimization results

C. Simulation and Experimental Results

In Fig. 8 the results obtained with the two methods are shown together. In [7] a method for comparing results from multi-objective optimizations is proposed, but since it involves knowledge of the Pareto-optimal surface we cannot apply it here. It is important to note that to obtain the 52 points with the aggregation method, a total number of $52 \times 200 \times 20 = 208000$ closed-loop simulations had

to be performed, while for the 50 points on the multi-objective method only $50 \times 200 = 10000$ simulations were needed. This means that MOGA is 20 times faster than the aggregation technique. Furthermore, the aggregation method was not able to find all points of the Pareto optimal set.

Three of the solutions, numbered from 1 to 3 in Fig. 8 were chosen and experiments on TINA were performed with the corresponding controllers. Points number 1 and 3, were chosen to match both the results from single-objective and multiobjective optimization. The simulation and experimental results are compared in Fig. 9 to 11.
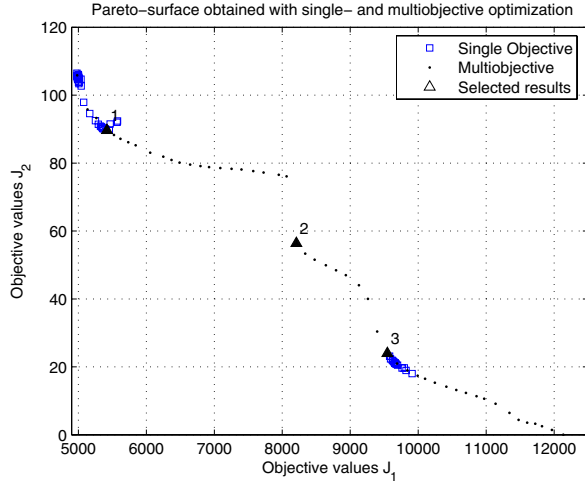

Fig. 8. Single and multi-objective optimization Pareto-sets and chosen points for experiments
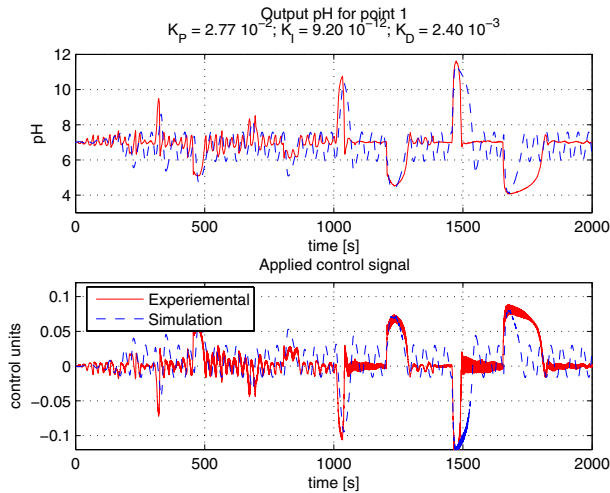

Fig. 9. Simulation and experimental results for point 1

As shown in the figures the integral gain of the PID controller is relatively small, because the plant itself has integral action: even a small input, applied to the plant for a long time, can drive it into saturation zones. The derivative gains are small, due to the oscillatory behavior introduced by modelling errors.
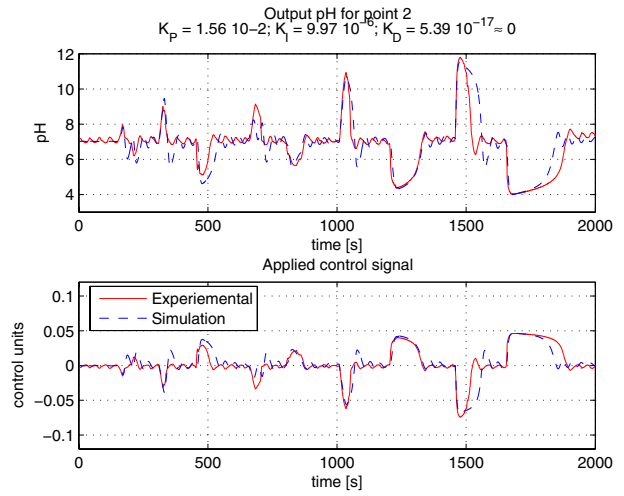

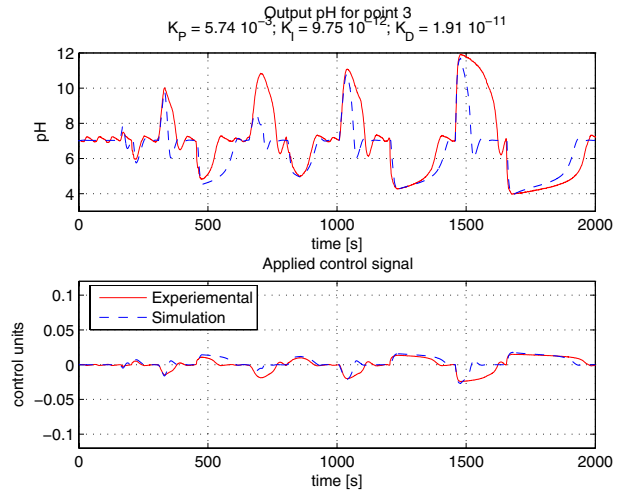
Fig. 10. Simulation and experimental results for point 2

Fig. 11. Simulation and experimental results for point 3

V. GAIN-SCHEDULED PID CONTROL

Having shown that multiobjective GA optimization delivers better results than the single-objective optimization for the given design problem, we will use the former to find a gain-scheduled PID controller for the titration and neutralization plant.

The objective functions remain the same as in (2) and (3). The gain-scheduled control law is of the form

$$K_P(k) = K_{P0} + K_{P1}\varepsilon(k) + K_{P2}\varepsilon^2(k) \qquad (4)$$

where $\varepsilon(k) = pH(k) - C_{pH}$. Thus we have a total of 6 design parameters: K_{P0}, K_{P1}, K_{P2}, C_{pH} and K_I and K_D.

Running SPEA with the same parameters (population size of 50 and 200 generations) we obtain a new Pareto-surface, which is shown in Fig. 12 together with the one obtained via a fixed-gain PID controller.

¿From the obtained points we select one (number 4 in Fig. 12) with coefficients
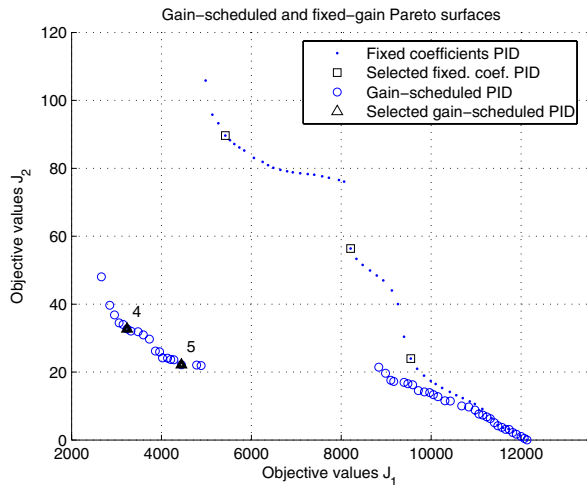
Fig. 12. Comparison between fixed-gain and gain-scheduled controllers

$K_{P0} = 5.26 \cdot 10^{-4}$; $K_{P1} = -2.52 \cdot 10^{-3}$; $K_{P2} = 1.25 \cdot 10^{-3}$; $C_{pH} = 8.68$; $K_I = 2.28 \cdot 10^{-8}$; $K_D = 2.62 \cdot 10^{-4}$.

To demonstrate the performance improvement achieved by gain-scheduling, the responses achieved with controllers corresponding to points 3 and 4 are compared in Fig. 13.
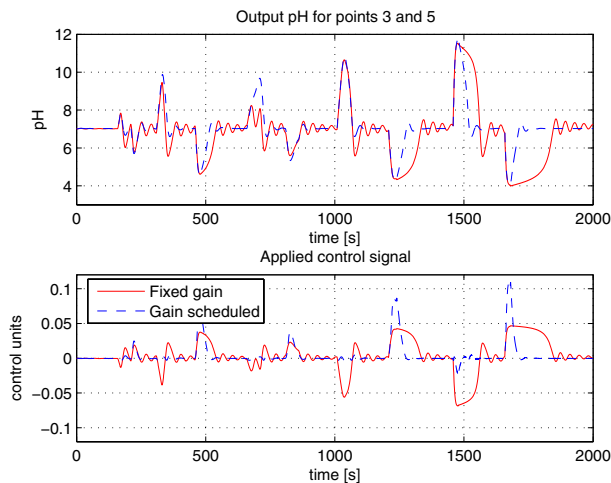

Fig. 13. Comparison of the response with fixed-gain and gain-scheduled PID control for the same value of J_2

¿From the results it is obvious that the gain-scheduled PID controller shows most of the time a faster disturbance rejection than the fixed-gain PID controller. For the last three disturbance impulses the response is almost two times faster with the gain scheduled controller. The only exception is the third group of disturbance impulses (600 to 900 s), where the fixed-gain PID controller shows better performance. The better performance of the gain-scheduled controller in rejecting acid disturbances is due to the larger control signal (larger amplitude for a short period of time).

VI. CONCLUSIONS AND FUTURE WORKS

Two GA search techniques - aggregation into a single objective and using Pareto-optimal solutions (MOGA) - have been used to design a fixed-gain PID controller for a highly nonlinear chemical pilot plant. Experimental results confirm that MOGA is superior in terms of efficiency. This latter approach has also been used to design a gain-scheduled PID controller for this plant, which has been successfully tested experimentally.

REFERENCES

[1] M. Mwembeshi, C. Kent and S Salhi, "Flexible on-line Modeling and Control of pH in waste neutralisation reactors", *Chemical Engineering & Technology Journal*, 27 (2), (2004), pp. 130-138.
[2] M. Mwembeshi, C. Kent and S Salhi, "A GA Approach to Robust and Flexible Modelling and Control of PH in reactors", *Computers and Chemical Engineering*, 28 (9), (2004), pp. 1743-1757.
[3] A. Farag, H. Werner, M. Aten, Decentralised Control of a High Voltage DC System Using Genetic Algorithms, *Proceeding of IFAC World Congress*, Barcelona, Spain,2002.
[4] D. Dougherty and D. Cooper, "A Practical Multiple Model Adaptive Strategy for Multivariable Model Predictive Control", *Control Engineering Practice*, 11 2003, pp. 649.
[5] K. Åström and T. Hägglund, *PID Controllers: Theory, Design and Tunning*, ISA-The Instrumentation, Systems and Automation Society, NC, 1995.
[6] C. Fonseca and P. Fleming, "Genetic algorithms for multiobjective optimization: Formulation, discussion and generalization", *Proceedings of the Fifth International Conference on Genetic Algorithms*, 1993, pp. 416423.
[7] E. Zitzler, K. Deb, and L. Thiele, "Comparison of multiobjective evolutionary algorithms: Empirical results", *Evolutionary Computation*, 8(2), 2000, pp. 173195.
[8] E. Zitzler, M. Laumanns and L. Thiele, "SPEA2: Improving the Strength Pareto Evolutionary Algorithm for Multiobjective Optimization", *Evolutionary Methods for Design, Optimisation and Control with Application to Industrial Problems*, Proceedings of the EURO-GEN2001 Conference, 2001, pp.95-100.
[9] M. Nørgaard, (on-line) http://www.iau.dtu.dk/research/control/nnsysid.html
[10] H. Pohlheim, "Visualization of Evolutionary Algorithms - Set of Standard Techniques and Multidimensional Visualization", *Proceedings of the Genetic and Evolutionary Computation Conference*, 1999, pp. 533-540.
[11] G. Agrawal, K. Lewis, K. Chugh, C.-H. Huang, S. Parashar and C. Bloebaum, "Intuitive Multidimensional Visualization for MDO", *10th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, 2004.
[12] I. Griffin, P. Schroder, A. Chipperfield, and P. Fleming, Multi-objective optimization approach to the ALSTOM gasifier problem, *Proceedings of the Institution of Mechanical Engineers*, 214(I), 2000, pp. 453-468.
[13] S. Bleuler, M. Laumanns, L. Thiele, E. Zitzler, "PISA - A Platform and Programming Language Independent Interface for Search Algorithms", *Conference on Evolutionary Multi-Criterion Optimization*, 4/2003, pp 494-508.