A RELAXATION-BASED HEURISTIC FOR THE DESIGN
OF COST-EFFECTIVE ENERGY CONVERSION SYSTEMS*

T. Ahadi-Oskui‡, F. Cziesla and G. Tsatsaronis
TU-Berlin, Institute for Energy Engineering,
Marchstr. 18, 10587 Berlin, Germany

H. Alperin and I. Nowak
HU-Berlin, Department of Mathematics,
Unter den Linden 6, 10099 Berlin, Germany

ABSTRACT

Mathematicians and engineers have developed in a joint research project a solution approach for performing simultaneous structural and parameter optimization in the design of cost-effective complex energy conversion systems. The paper presents a methodology and an application to the design of a combined-cycle power plant that provides fixed amounts of electricity and steam for a paper factory. A superstructure that has embedded several potential configurations of such a cogeneration system is used to minimize the total cost of the plant products. The design problem is formulated as a nonconvex mixed-integer nonlinear program (MINLP), and solved via a rounding heuristic based on an automatically generated convex relaxation of the given problem. Utilities and the process industry are expected to benefit from such a MINLP optimization technique.

Keywords: Cost minimization, cogeneration plant, mixed integer nonlinear optimization, convex relaxation, rounding heuristic

INTRODUCTION

The optimization of the design of energy conversion systems has been studied extensively by engineers for many years now. Mathematical programming techniques have been of growing interest in the last years in addition to heuristic methods such as the thermoeconomic analysis [23, 17, 21, 22], and stochastic approaches such as evolutionary algorithms [9, 4, 18].

Different mathematical programming approaches have been developed [12], and have been used for the optimization of process synthesis problems [11, 13, 16]. Only few publications deal with mathematical programming approaches applied to the optimization of energy conversion systems [6, 20].

In a joint research project of mathematicians and engineers, a new approach has been developed to optimize nonconvex MINLP-problems resulting from a superstructure of a complex energy conversion system. The proposed solution method is based on a convex relaxation of the problem, which is constructed through a series of operations. Starting from an automatically generated block-separable reformulation of the problem, all nonconvex functions are replaced by convex underestimators. The resulting nonlinear convex relaxation is improved by an outer approximation algorithm, and finally solution candidates are computed via a rounding heuristic.

A main advantage of the proposed approach is the use of decomposition, i.e. the generation of convex underestimators and cuts are based on small-scale optimization problems. Furthermore, the optimal value of the relaxation is a lower bound on the optimal value of the original optimization problem. It can be used as a quality measure for the obtained solutions.

*The work is supported by the German Research Foundation (DFG) under grants NO 421/2-1 and TS 79/1-1.
‡Corresponding author, Phone: +49-30-314-23281 Fax: +49-30-314-21683 E-mail: oskui@iet.tu-berlin.de
The optimization method is coded in C++, and is part of a software package called LaGO for MINLP optimization [19]. Preliminary numerical results for our design problem are reported here. The use of relaxation-based methods for solving practical relevant large-scale MINLPs is quite new [7], and the integration of the two well established areas, nonlinear and mixed integer optimization, does not belong to the “traditional” operation research areas yet. It is a main issue of this paper to further this integration for solving complex design problems.

MODEL DESCRIPTION
The goal of the optimization is to design an energy conversion system for a paper factory with minimum total levelized costs per time unit. Parameter and structural changes are considered for the optimization simultaneously. A simple superstructure of a combined cycle power plant was developed as the basis for the optimization. The superstructure (Figure 1) consists of a gas turbine as topping cycle and a subsequent heat recovery steam generator (HRSG) that supplies a steam turbine as bottoming cycle. The process steam is extracted before it enters the low-pressure steam turbine. There is a total of five structural variables in the superstructure. The first structural variable refers to the selection of the gas turbine system among two different types (GE Frame 6/39.1 MW or Siemens V 64.3/67.5 MW). The second and third structural variables determine whether there are additional duct burners in the HRSG. The possibility of reheating the steam after the high-pressure steam turbine is introduced by the fourth structural variable. The last of these variables determines if there is additional steam generation and superheating at intermediate pressure. A complete combustion is assumed in all components in which combustion takes place.

The required demand of 90 MW electric power and 99.5 t/h process steam at 4.5 bar refers to a real paper factory [3]. The cogeneration plant has to fulfill primarily the needs for thermal energy of the paper machines. If more electricity is produced than required, the excess is sold on the market; in the opposite case, the deficit is bought from the network. The models of the plant components are chosen in a way that they facilitate the work of the solver LaGO on the one hand and provide reasonable accuracy on the other hand. Besides the mandatory mass- and energy balances almost every plant component model contains one or more characteristic equations. Some examples are:

**Gas turbine compressor [5]:**
\[ \eta_{is} = \frac{h_s - h_i}{h_s - h_e} \]

\( \eta_{is} \): isentropic efficiency, \( h_s \): enthalpy after isentropic compression, \( h_i \): inlet enthalpy, \( h_e \): exit enthalpy.

**Steam turbine [27]:**
\[ \eta_{pol} = 1 - \frac{s_j - s_e}{s_{is} - s_e} \]

\( \eta_{pol} \): polytropic efficiency, \( s_{is} \): entropy after isenthalpic expansion, \( s_j \): inlet entropy, \( s_e \): exit entropy.

**Heat exchanger:**
\[ A = \frac{\dot{Q}}{k \cdot \Delta T_{ln}} \]

\( A \): Heat exchanger area, \( \dot{Q} \): heat transfer rate, \( k \): overall heat transfer coefficient, \( \Delta T_{ln} \): logarithmic mean temperature difference.

A complete combustion is assumed in all components in which combustion takes place.

The cost functions for the plant components are either taken from the literature [25, 24, 1] or are developed from available cost data [10]. The total levelized costs of the plant design are calculated using the TRR-method from [5] and, depending on the case, adding the levelized costs of purchased electric power or subtracting the levelized revenues of sold electric power.

Regarding the property equations, all gases, including the water vapor in the exhaust gas stream, are treated as ideal gases. The state variables are calculated with polynomial functions from [15]. The state variables of water and steam in the water/steam cycle are calculated with the IAPWS-IF97 formulation [26] which represents the industrial
standard.

For the optimization with the developed software package LaGO the entire model has to be written as one system of equations. Therefore, the approach is different from that of sequential modeling. There are neither jumps in the model nor distinction of cases possible. All equations are active throughout the optimization. Every variable has to be defined within valid boundaries. In addition, some of the model equations were reformulated to increase the efficiency and robustness of the LaGO algorithm. For instance, one of the equations to calculate the saturation temperature

\[ D = \frac{2G}{-F - (F^2 - 4EG)^{0.5}} \]

was transformed to

\[ x = D^2 \\
Ex + FD + G = 0 \\
F^2 - 4EG \geq 0 \\
2DG + xF \leq 0 \]

using only quadratic and bilinear functions. Hereby, E, F and G are functions of the pressure.

All in all the model contains 508 variables and 461 constraints. The optimization problem can be described verbally by:

**Minimize:** Total levelized costs per time unit

**Subject to:** constraints referring to

- plant components
- material properties
- investment, operating and maintenance cost
- economic analysis

**MINLP-OPTIMIZATION**

The previously described design problem can be formulated as the following mixed integer nonlinear program (MINLP):

\[
\text{min} \quad f(x, y) \\
\text{s.t.} \quad g(x, y) \leq 0 \\
h(x, y) = 0 \\
x \in [x_l, x_u] \\
y \text{ binary.}
\]

Since some of the thermodynamic and economic equations are nonlinear, problem (P) is nonconvex, and, therefore, cannot be solved with traditional convex MINLP methods. In the following, we describe a heuristic for solving (P), which is based on convexifying the problem.

**An illustrative example**

In order to explain the basic ideas of the mathematical method, we consider the following simple MINLP:

\[
\min \{ x_2 \mid g_1(x) \leq 0, \quad g_2(x) \leq 0, \quad x_1 \in \{0, 1\}, \quad x_2 \in [0, 1] \},
\]

where \( g_1 \) is a complicated nonconvex function and \( g_2 \) is a convex function. The proposed solution method consists of the following four steps, which are illustrated in Figure 2. In the first step, a convex relaxation is constructed by replacing \( g_1 \) with a quadratic convex underestimator \( q_1 \) and relaxing the binary constraint by \( x_1 \in [0, 1] \). The resulting convex NLP is solved obtaining the point \( x_1 \). In the second step, a polyhedral relaxation is generated by a linearization \( \tilde{g}_2 \) of \( g_2 \) at \( x_1 \) and by a supporting hyperplane \( \tilde{g}_1 \) of \( g_1 \) which is orthogonal to \( \nabla q_1(x_1) \). The polyhedral relaxation is solved obtaining a point \( x^2 \). In the third step, the binary component of \( x^2 \) is rounded to \( x^3 \), and the polyhedral relaxation with fixed binary variables is solved obtaining a point \( x^4 \). Finally, a local minimizer \( x^5 \) of the original problem with fixed binary variables is computed using \( x^4 \) as a starting point. We describe the steps of this method now more detailed.

**Block-separable reformulation**

Due to the component structure of the given optimization problem, all functions of (P) can be represented as a sum of sub-functions, which depend on a small number of variables. Optimization problems having such a property are called block-separable, and can be reformulated in the following form:
differentiable. Here, we used the following notation. \( IR \) is the index set of binary variables, \( \{1, \ldots, n\} \) is the index set of constraints that contain variables in block \( k \), \( B \subseteq \{1, \ldots, n\} \) is the index set of binary variables, \( c \in \mathbb{R}^n \), \( b \in \mathbb{R}^m \), \( A \in \mathbb{R}^{(m, n)} \), and the functions \( g_i^k \) are twice-differentiable. Here, we used the following notation. For an index set \( J \subseteq \{1, \ldots, n\} \), we denote by \( |J| \) the number of elements of \( J \). A subvector \( x_J \in \mathbb{R}^{|J|} \) of \( x \in \mathbb{R}^n \) is defined by \( (x_J)_{i\in J} \).

The generation of problem (S) from problem (P) is done automatically using a sampling technique (see [19]). Formulation (S) supports decomposition: Since all nonlinear constraint functions \( g_i^k \) of (S) depend on a small number of variables, we can generate a convex relaxation of (S) by constructing low-dimensional underestimators and cuts.

**Convex relaxation**

A convex relaxation of problem (S) is defined by

\[
\begin{align*}
\min & \quad c^T x \\
\text{s.t.} & \quad Ax + b \leq 0 \\
 & \quad g_i^k(x_{J_k}) \leq 0, \quad i \in M_k, \ k = 1, \ldots, p \\
 & \quad x \in X, \ x_B \text{ binary}
\end{align*}
\]

where \( q_i^k \) are convex underestimators of \( g_i^k \) over \( X_k = [x_{J_k}, x_{J_k}] \), i.e. \( q_i^k(x) \leq g_i^k(x) \) for all \( x \in X_k \) and \( q_i^k \) is convex over \( X_k \). The best convex underestimators are convex envelopes. Since computing convex envelopes of nonconvex functions can be computationally very expensive, we replace all nonconvex functions by nonconvex quadratic underestimators using a sampling technique (see [19]). Finally, all nonconvex quadratic functions are replaced by convex quadratic \( \alpha \)-underestimators introduced by Adjiman and Floudas [2].

**Outer approximation**

Computational experiments showed that for some convex underestimators \( q_i^k \) of (C) the approximation error is quite large. In order to improve the relaxation, a cutting plane method was developed, which generates polyhedral relaxations of the form:

\[
\begin{align*}
\min & \quad c^T x \\
\text{s.t.} & \quad Ax + b \leq 0 \\
 & \quad (c_i^k)^T x_{J_k} + d_i^k \leq 0, \quad i \in M_k, \ k = 1, \ldots, p \\
 & \quad x \in X.
\end{align*}
\]

In addition to being stronger than the nonlinear relaxation (C), the linear relaxation (R) has the advantage, that it can be solved much faster than (C). The following cutting plane algorithm for generating (R) is a generalization of Kelley’s cutting plane algorithm [14].

**Algorithm 1 (cutting plane algorithm)**

1. Initialize (R) by setting \( M_k = \emptyset, \ k = 1, \ldots, p \) and \( X \) as the smallest box containing the feasible set of (C).

2. Compute a solution \( \hat{x} \) of (R), and for each block \( k \) add some cuts \( (c_i^k)^T x_{J_k} + d_i^k \leq 0 \) to (R) which are violated at \( \hat{x} \), by adding the index \( i \) of the violated constraint to \( M_k \).

3. If the iteration limit is not exceeded and \( \hat{x} \) is changed significantly, goto 2.

We use two types of cuts. The first cuts are based on linearizing the convex constraints \( q_i^k(x_{J_k}) \) at the solution point of the convex sub-problem.
\[
\begin{align*}
\text{(C)} & \\
\min & \quad a_k^T x \\
\text{s.t.} & \quad q_k^T(x) \leq 0, \quad i \in M_k \\
& \quad x \in X_k,
\end{align*}
\]

where
\[
da_k = - \sum_{i \in V^k} \nabla q_i^k(\hat{x}_j) \cdot \| \nabla q_i^k(\hat{x}_j) \|^{-1}
\]

and \(V^k\) is the index set of convexified constraints of \((C)\), which are violated at \(\hat{x}\). Denote by \(\hat{x}^k\) a solution point of \((C_k)\). By linearizing the active constraints of \((C_k)\) at \(\hat{x}^k\) we obtain the following valid cuts
\[
\nabla q_i^k(\hat{x}^k)^T (x_j - \hat{x}^k) \leq 0, \quad i \in A^k
\]

where \(A^k = \{i \in M_k \mid q_i^k(\hat{x}^k) = 0\}\) is the index set of active constraints.

The second type of cuts are used, if the convex underestimator \(q_i^k\) is a bad approximation of a nonconvex function \(g_i^k\). In this case, we solve the problem
\[
\max \{ (c_i^k)^T x_j \mid x_j \in [\underline{x}_j, \overline{x}_j], \quad g_i^k(x_j) \leq 0, \quad x_j \text{\binary}\}
\]

where \(c_i^k = \nabla q_i^k(\hat{x}_j)\). Let \(t_i^k\) be the optimal value of (1). Then
\[
(c_i^k)^T x_j \leq t_i^k
\]
is a valid cut. For solving (1), we use a sampling method.

**Rounding heuristic**

For computing solution candidates of problem (P), a rounding heuristic was developed, which is based on the convex relaxation \((C)\) or \((R)\) [19]. The heuristic works by computing a solution point of problem \((C)\) or \((R)\), and rounding some binary components of this point. The rounded variables are fixed and the restricted convex relaxation with fixed binary variables is solved. This procedure is repeated as long as either the restricted convex relaxation is infeasible or all binary variables are fixed. In the last case a local search is started from the last solution of the restricted convex relaxation giving a solution candidate. The values of the binary variables are recursively switched and the whole process is repeated as long, as either all combinations of binary variables are searched, or the number of solution candidates exceeds a given number.

**RESULTS**

The described algorithm was coded in C++ as part of the software package LaGO [19]. In addition, a Matlab formulation of the model was developed to apply the Matlab optimization toolbox FMINCON [8] for finding local solutions of the problem. To analyze the performance of the proposed algorithm, the code was run on a machine with a 1GHz Pentium III processor and 256 MB RAM.

In order to improve the sampling method for computing quadratic underestimators of nonconvex functions, a simple base design of the cogeneration plant was integrated in the sample set. Therefore a simple structure of the plant and moderate values for the continuous decision variables were chosen. The variables of the base design are shown in the second column in table 2 (Base). In the Table, a value of zero represents an inactive variable. The base design produces 72 MW electric power and has a total levelized cost of 6823 Euro/h. Based on the binary and continuous decision variables, a complete starting point for the model was generated, i.e. an initial value for every variable in the model was computed. Again, the Matlab formulation of the superstructure was used for that purpose. Since this formulation depends on the decision variables only, it calculates the values of all dependent variables for a given set of decision variables.

After running the proposed optimization method, a block-separable reformulation \((S)\) with \(p = 172\) blocks is generated, where the largest block-size is \(\max |J_k| = 47\). For the generation of the convex relaxation \((C)\), 524 convex underestimators of nonconvex functions are constructed. The solution of \((C)\) gives the lower bound 5547.13 Euro/h for the total levelized cost. Rounding the structural variables of this solution and performing a local optimization results in a design with a total levelized cost of 6090.80 Euro/h and an electric power generation of 85.69 MW. The corresponding decision variables are shown in column three of Table 2 (LaGO).

Since the problem is of small size regarding the binary variables local optimizations of all 32 possible structural variable combinations are performed for comparison. For this, 50 randomly created points for every binary combination are used as starting points for a local optimization of the remaining continuous decision variables. The best found design has a total levelized cost of 5995.83 Euro/h and generates...
90 MW of electric power. The parameters of that design are shown in the fourth column of Table 2 (LO).

Considering the preliminary status of LaGO, the results are promising. There is only a difference of 1.6% compared to the best solution found by complete exploration of the binary space combined with a multistart local optimization which is only reasonable for a combinational problem of this small size. The comparison of the last two columns of Table 2 shows that the nearly optimal solutions could be significantly different. However, the difference in the value of the objective function between such solutions is in general relatively low.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Base</th>
<th>LaGO</th>
<th>LO</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gas turbine</td>
<td>Frame6</td>
<td>V63.4</td>
<td>V63.4</td>
</tr>
<tr>
<td>1. Duct Burner</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>2. Duct Burner</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Reheating</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Add. Steam</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>$\dot{m}_p$ [kg/s]</td>
<td>40</td>
<td>30.03</td>
<td>32.18</td>
</tr>
<tr>
<td>$\dot{m}_S$ [kg/s]</td>
<td>0</td>
<td>0</td>
<td>5.40</td>
</tr>
<tr>
<td>$p_{hp}$ [bar]</td>
<td>70</td>
<td>113.80</td>
<td>63.46</td>
</tr>
<tr>
<td>$T_{hp}$ [K]</td>
<td>825</td>
<td>784.1</td>
<td>812.6</td>
</tr>
<tr>
<td>$p_{rp}$ [bar]</td>
<td>0</td>
<td>0</td>
<td>53.73</td>
</tr>
<tr>
<td>$T_{rp}$ [K]</td>
<td>0</td>
<td>0</td>
<td>771.0</td>
</tr>
<tr>
<td>$p_{ip}$ [bar]</td>
<td>4.5</td>
<td>5.82</td>
<td>4.5</td>
</tr>
<tr>
<td>$T_{ip}$ [K]</td>
<td>0</td>
<td>0</td>
<td>479.3</td>
</tr>
<tr>
<td>$\Delta T_{hp}$ [K]</td>
<td>15</td>
<td>5.0</td>
<td>5.4</td>
</tr>
<tr>
<td>$\Delta T_{ip}$ [K]</td>
<td>15</td>
<td>7.0</td>
<td>24.2</td>
</tr>
<tr>
<td>$\eta_{HP}$</td>
<td>0.850</td>
<td>0.765</td>
<td>0.750</td>
</tr>
<tr>
<td>$\eta_{MP}$</td>
<td>0</td>
<td>0</td>
<td>0.755</td>
</tr>
<tr>
<td>$\eta_{LP}$</td>
<td>0.850</td>
<td>0.762</td>
<td>0.923</td>
</tr>
<tr>
<td>$\dot{n}_{B1}$ [kmol/s]</td>
<td>0.0800</td>
<td>0</td>
<td>0.0097</td>
</tr>
<tr>
<td>$\dot{n}_{B2}$ [kmol/s]</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Total lev. cost</td>
<td>6823</td>
<td>6090.80</td>
<td>5995.83</td>
</tr>
<tr>
<td>[Euro/h]</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

CONCLUSIONS

We presented a MINLP solution method for optimizing the design of a cost-effective complex energy conversion method. In contrast to sampling methods, the proposed solution approach generates a convex relaxation, which provides a quality measure on the obtained solutions. Furthermore, the approach supports decomposition, and it offers the possibility to improve the relaxation by adding cuts. Preliminary numerical results show that the obtained convex relaxation is reasonable. Future research will focus on the improvement of the relaxation using deeper cuts, and on the development of a branch-and-bound algorithm for reliable global optimization.

REFERENCES


Figure 1: Simple superstructure of the cogeneration plant

Table 1: Continuous Decision Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Symbol</th>
<th>Bounds</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>mass flow rate of the high-pressure steam</td>
<td>$\dot{m}_p$</td>
<td>[30, 100] kg/s</td>
<td></td>
</tr>
<tr>
<td>mass flow rate of steam generated at intermediate-pressure (if existing)</td>
<td>$\dot{m}_S$</td>
<td>[0, 20] kg/s</td>
<td></td>
</tr>
<tr>
<td>pressure at the inlet of the steam turbine HP</td>
<td>$p_{hp}$</td>
<td>[30, 160] bar</td>
<td></td>
</tr>
<tr>
<td>temperature at the inlet of the steam turbine HP</td>
<td>$T_{hp}$</td>
<td>[700, 870] K</td>
<td></td>
</tr>
<tr>
<td>pressure at the outlet of steam turbine HP if reheating exists</td>
<td>$p_{rp}$</td>
<td>[10, 60] bar</td>
<td></td>
</tr>
<tr>
<td>temperature of the reheated steam (if existing)</td>
<td>$T_{rp}$</td>
<td>[700, 870] K</td>
<td></td>
</tr>
<tr>
<td>pressure of the intermediate-pressure steam</td>
<td>$p_{ip}$</td>
<td>[4.5, 10] bar</td>
<td></td>
</tr>
<tr>
<td>temperature of the intermediate-pressure steam</td>
<td>$T_{ip}$</td>
<td>[430, 530] K</td>
<td></td>
</tr>
<tr>
<td>subcooling in the high-pressure economizer</td>
<td>$\Delta T_{hp}$</td>
<td>[5, 25] K</td>
<td></td>
</tr>
<tr>
<td>subcooling in the intermediate-pressure economizer (if existing)</td>
<td>$\Delta T_{ip}$</td>
<td>[5, 25] K</td>
<td></td>
</tr>
<tr>
<td>polytropic efficiency of the steam turbine HP</td>
<td>$\eta_{HP}$</td>
<td>[0.75, 0.95]</td>
<td></td>
</tr>
<tr>
<td>polytropic efficiency of the steam turbine MP (if existing)</td>
<td>$\eta_{MP}$</td>
<td>[0.75, 0.95]</td>
<td></td>
</tr>
<tr>
<td>polytropic efficiency of the steam turbine LP</td>
<td>$\eta_{LP}$</td>
<td>[0.75, 0.95]</td>
<td></td>
</tr>
<tr>
<td>molar flow rate of fuel to the first duct burner in the HRSG (if existing)</td>
<td>$\dot{n}_{B1}$</td>
<td>[0, 0.5] kmol/s</td>
<td></td>
</tr>
<tr>
<td>molar flow rate of fuel of the second duct burner in the HRSG (if existing)</td>
<td>$\dot{n}_{B2}$</td>
<td>[0, 0.5] kmol/s</td>
<td></td>
</tr>
</tbody>
</table>