



Article A Hybrid Metaheuristic Optimization Approach for the Synthesis of Operating Procedures for Optimal Drum-Boiler Startups

Emilio Garduno¹, Erik Rosado-Tamariz^{1,2,*}, Miguel A. Zuniga-Garcia^{1,2}, and Rafael Batres^{1,*}

- ¹ School of Engineering and Sciences, Tecnologico de Monterrey, Autopista Mexico-Acapulco, Col. Real del Puente, Cuernavaca 62790, Mexico; emilio.gh@tec.mx (E.G.); miguel.angel.zuniga@ineel.mx (M.A.Z.-G.)
- ² Instituto Nacional de Electricidad y Energías Limpias (INEEL), Av. Reforma 113, Col. Palmira, Cuernavaca 62490, Mexico
- * Correspondence: erosado@ineel.mx or erik.rosado@tec.mx (E.R.-T.); rafael.batres@tec.mx (R.B.)

Abstract: A steam generator serves as a power generation equipment that uses the expansive power of steam to generate electricity. The startup process of a steam generator plays an important process in a power plant to adjust its electricity generation in response to changes in load demand. As renewable generation plants increase, the levels of variability in electricity production increase. Fast startups become instrumental as they enable traditional power generation plants to provide the quantity of electricity missing when variable renewable energies cannot satisfy the load demand. The drum boiler is one of the main pieces of equipment involved in the startup process of a steam generator. However, if the startup process is carried out too fast, excessive thermal stresses may occur, thus provoking damage to the components of the drum boiler. This paper proposes a dynamic optimization methodology to synthesize operating procedures that minimize the startup time of the drum boiler while avoiding the excessive formation of thermal stresses. Since valve operations influence the time-varying behavior of the steam, dynamic simulation is needed in order to evaluate the operating procedure. The proposed algorithm is based on two important elements of two metaheuristic algorithms: the acceptance probability of the simulated annealing algorithm and the tabu search memory structures. A case study evaluates the proposed approach by comparing it against results previously published in the literature.

Keywords: hybrid metaheuristic optimization; simulated annealing; tabu search; steam generation process; thermal power plants; synthesis of operating procedures

1. Introduction

Conventional thermal power plants (CTPP) play a key role to deal with one of the biggest challenges of the energy sector: the reliable and efficient supply of electricity. The CTPP provides backup energy to the electric power system to balance the variable electricity demand and the intermittent generation of non-conventional renewable energy sources such as wind and solar energy. When coping with normal demand variations or when variable renewable energies cannot meet the demand, CTTP generation has to be adjusted employing lower or greater production of electricity (reduction or increase of load) respectively. This adjustment is also applied by doing start-ups and shutdowns of equipment in the power plants. In this context, rapid startups and shutdowns improve the operational flexibility of the power plant by adding to it competitive advantage [1–4]. Therefore, the optimal design of operating profiles is a research area with high potential [5–8].

A thermal power plant often uses a steam generator to take advantage of the heat obtained from its main electricity generation process. In a steam generator, water is heated and then turned into steam that spins a turbine, which is connected to a generator that produces electricity.



Citation: Garduno, E.; Rosado-Tamariz, E.; Zuniga-Garcia, M.A.; Batres, R. A Hybrid Metaheuristic Optimization Approach for the Synthesis of Operating Procedures for Optimal Drum-Boiler Startups. *Appl. Sci.* 2021, *11*, 5085. https://doi.org/10.3390/ app11115085

Academic Editor: Alberto Benato

Received: 3 April 2021 Accepted: 26 May 2021 Published: 30 May 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https:// creativecommons.org/licenses/by/ 4.0/). To create high-temperature, high-pressure steam in the steam generator, fuel energy is converted to heat, which is transferred to the drum boiler.

Figure 1 shows a typical drum boiler. As explained by Aström and Bell [9] the drum boiler has a reservoir for water and steam with a water inlet and a steam outlet at the top. The drum stores the steam generated in the water tubes and acts as a phase-separator for the steam-water mixture. The difference in densities between hot and cold water along with the gravity contributes to the accumulation of the "hotter" water and saturated steam into the drum boiler.



Figure 1. Schematic representation of the drum boiler.

The drum boiler has the potential to improve the competitiveness of a thermal power plant by reducing its startup time. The rate at which a boiler can be brought up to a normal operating state depends on its size, and the length of time it has been shut down. In general, the larger and colder a boiler, the longer it takes to startup.

The startup is carried out by operating the control valves in ways that the steam flow rate and pressure increase up to their normal setpoints when the steam pressure and combustion control system can be switched over to automatic.

Although fast startups improve the competitiveness in an open power market, if a startup is carried out too fast, excessive thermal stress can occur in the drum boiler components [10]. Therefore, feasible operating procedures must consider the physical constraints of the drum boiler that ensure its integrity.

Operating procedure synthesis (OPS) can be described as a planning problem, where actions and their sequence must be found in order to take the process from an initial state to a goal state, such as in startups and shutdowns [11]. Additionally, transient operations can take advantage of OPS in terms of both safety and cost [12].

Despite the focus on startup-time reduction, current approaches [9,13,14] fail shortly because they can only obtain startup curves of the drum boiler state variables but cannot identify the corresponding control actions (operations) and their sequence.

This paper is structured as follows: Section 2 contains the literature review. Section 3 the problem description. In Section 4 the proposed approach is presented. Next, the case study is described in Section 5. Then, in Section 6 the experiments and results are presented. Finally, Section 7 presents the conclusions and future work.

2. Literature Review

OPS can be considered as a search for a set of sequenced primitive operations that transform a plant system from an initial state to a pre-specified goal state through a series of intermediate states. These primitive operations must be carried out in such a way that no violations are made of any relevant process or mechanical, safety, and environmental constraints. Most attempts to solve OPS problems have relied on simplified process behavior models [12]. In contrast to this, simulation-based planning approaches make use of detailed dynamic behavior models of the process and a mathematical representation of quantitative constraints embedded within a rigorous dynamic optimization framework.

Regarding drum boiler transient operations, a lot of work has been done regarding the optimization of steam generation in a drum boiler from a procedural point of view. Aström and Bell [9] developed a nonlinear physical model with a complexity that is suitable for dynamic optimization and OPS. The model is based on physical parameters for the plant and can be easily scaled to simulate any power plant drum boiler.

Franke et al. [15] developed a nonlinear dynamic model of a drum boiler based on Aström's physical model, using the Modelica language. Their model had three control inputs in terms of feedwater flow rate, heat supply, and steam outlet. They solved a dynamic optimization problem using a sequential quadratic programming (SQP) algorithm. Using this approach, the startup time could be reduced by 30%.

Kruger et al. [16] proposed a quadratic programming optimization approach to determine the optimal values of steam pressure and steam temperature in a startup process. Their model considered hard constraints such as control bounds and stress levels for the drum and header. They concluded that their optimization model could minimize both fuel consumption and startup time.

Belkhir et al. [17] investigated the minimization of the startup time of a drum boiler. Their proposed startup strategy defined the initial and goal states in terms of steam mass flow rate and the internal pressure of the drum. The startup process was formulated as an optimal control problem that minimized a quadratic objective function under physical and operational constraints. The physical constraints were related to the structural integrity of thick-walled components due to higher thermal stresses. The optimization problem was solved by combining a framework developed on the JModelica environment and the interior-point optimizer algorithm (IPOPT). Their results were compared with the optimal start-up trajectories in Franke et al. [18], and the optimized profiles reached desired states in a shorter time without violating the operational and physical constraints.

Zhang et al. [19] reported the dynamic analysis of the steam and water system of the natural circulation boiler using the environment of MATLAB/Simulink. They proposed a boiler model based on the work of Aström and Bell with specific parameters to simulate the dynamic analysis of the steam water system. They solved the model using the ode45 algorithm, which is based on the fourth-order Runge–Kutta and Dormand–Prince methods. The boiler startup aimed at saving water and fuel.

Nevertheless, these works have limited applicability since they were solutions to specific problems. For instance, in many cases, the simulation model is embedded within the optimization tool and it is not possible to scale them for more complex problems. Other works propose approaches using commercial tools for the coupling of a simulation-optimization system. The drawback is that these tools operate as black boxes, with limited information about the modeling assumptions. A third group of contributions, despite considering thermal stress evaluation, seek to minimize startup times regardless of how the plant must be operated to achieve a given goal state.

To overcome the limitations of previous works, this paper proposes a scalable approach for the synthesis of operating procedures that minimize the startup time of the drum boiler while avoiding the excessive formation of thermal stresses. The proposed approach is based on a dynamic optimization methodology with a hybrid-metaheuristic algorithm that generates the optimal startup procedure of a drum boiler. The proposed algorithm is based on two important elements of two metaheuristic algorithms. Namely, the search zone in the cooling element from the simulated annealing algorithm and the efficient computational performance provided from the tabu search algorithm memory structures. A case study evaluates the proposed approach by comparing it against results previously published in the literature.

3. Problem Description

The problem consists of finding the control valve actions and their sequence to minimize the time needed to take the system from an initial state to a goal state while satisfying mechanical and process constraints. This problem is formulated as a dynamic optimization problem, in which the objective function is stated in terms of the internal drum pressure and the outlet steam flow rate. To ensure mechanical integrity, the thermal stress must be kept within specified limits.

The drum boiler is a multi-input and multi-output system whose state variables that change over time with a non-linear behavior [20]. The drum boiler system (DBS) can be divided into two main subsystems, a water circulation loop, and a heat energy system [21]. The DBS is responsible for the production and regulation of saturated steam (main steam), which is sent to the superheaters to produce superheated steam which drives the steam turbine and generates electricity. The saturated steam amount and quality are controlled by adjusting the steam generator water level, steam flow, feedwater flow, and heat supply. The steam drum, mud drum, the downcomer water wall tubes, and the riser water wall tubes are the main parts of the water circulation loop. Whereas the heat energy system refers to a combustion process (furnace) for thermoelectric power plants or the gas turbine flue exhaust gases for a combined cycle power plant. The steam drum has the function of controlling the steam generator water level. Likewise, the mud drum has the function of a settling point for solids in the boiler feedwater. The downcomer water wall is the cooler water line that transfers water from the steam drum to the mud drum. The riser water walls are the hotter water line that contains boiler feedwater that is heated by radiant heat from the flue gas and boiled to produce steam that flows upward to the steam drum. The heat supplies from the flue gases to the water flowing down the riser water wall tubes to regulate the boiling process. A centrifugal pump supplies the feedwater to the steam drum, and the saturated steam sent to the superheaters is regulated through a control valve. The DBS structure based on the reported by Liu et al. [22] is shown in Figure 2.



Figure 2. Drum boiler system structure.

4. Problem Description

The proposed approach (Figure 3) integrates a simulator with an optimizer. The simulator uses a dynamic simulation model while the optimizer relies on a metaheuristic optimization algorithm. The system architecture is designed in such a way that a simulation model can be replaced without having to modify the optimization component. Similarly, the optimization algorithm can be replaced while keeping the same simulation model in the simulator.



Figure 3. System architecture.

4.1. Simulator

The dynamic behavior of the drum boiler is simulated by solving an ordinarydifferential-equation model. The simulation model represents the drum boiler in terms of a water inlet, a heat supply, a water-level PI-controller, and a saturated steam outlet. A schematic representation of the drum boiler is shown in Figure 4.



Figure 4. Schematic diagram of the drum boiler.

Water from the condenser enters the drum through the water inlet and the saturated steam is extracted. The behavior of the boiler furnace in a coal-fired power plant or exhaust gases of a gas turbine is modeled using a heat supply system to heat and evaporate the water in the rising tubes. For simplicity, the model assumes thermodynamic equilibrium

between water and steam inside the drum. The mass balance in the drum boiler can be written as:

$$\left(\frac{\partial\rho_l}{\partial p}V_l(l_l) + \frac{\partial\rho_v}{\partial p}V_v\right)\frac{dp}{dt} + (\rho_l - \rho_v)\frac{dV_l(l_l)}{dt} = q_l - q_v \tag{1}$$

where V_l is the water volume in the drum, V_v the steam volume in the drum, ρ_l and ρ_v the density of water and steam, l_l the water level inside the drum, p the pressure inside the drum, q_l feedwater flow and q_v the steam flow rate extracted from the drum.

The energy balance in the drum boiler can be expressed as:

$$(V_v(h_v\frac{\partial\rho_v}{\partial p} + \rho_v\frac{\partial h_v}{\partial p}) + V_l(h_l\frac{\partial\rho_l}{\partial p} + \rho_l\frac{\partial h_v}{\partial p}) - V + mC_p\frac{\partial Tv}{\partial p})(\frac{dp}{dt}) + (\rho_lh_l + \rho_vh_v)\frac{dV_l}{dt} = Q + (q_lh_l - q_vh_v)$$
(2)

where *Q* is the heat flow, *V* the water-steam volume in the drum, h_l and h_v are the water and steam enthalpies, while C_p denotes the specific heat capacity of steam.

According to Franke et al. [18], thermal stresses occur in the thick-walled drum if there are spatial temperature differences. Thus, thermal stress is determined proportionately to the time derivative of the metal temperature, considering that the metal temperature is equivalent to the water saturation temperature inside the drum, as given in equation three.

$$\sigma_D = k \frac{dT_D}{dt} \tag{3}$$

where σ_D is the thermal stress in the thick-walled drum, *k* the thermal conductivity of the wall and $T_D = T_{sat}(p)$ the inner temperature in the drum.

For simplicity, bulk system flows, volumes, and masses are considered. Therefore, the model ignores spatial variations in the process variables such as individual geometric features and fin and pipes arrangements in the risers and downcomers. Moreover, this model does not consider heat losses between the water inside the drum and the walls of the drum and pipes. For that reason, the water and metal temperatures are assumed to be in thermal equilibrium within the drum. Despite these simplifications, the resulting lumped parameter model could capture the overall behavior of the drum boiler.

4.2. Optimizer

The optimizer generates an operating procedure given the initial and the goal states. The optimizer starts by generating an initial feasible solution searching through the space of solutions. Then, the optimization algorithm iteratively improves the initial solution by making local changes until there is no better solution when applying such changes. Metaheuristic optimization algorithms involve the encoding of solutions, the manipulation of encoded solutions by operators, and a selection based on their objective function to find an optimum or near-to-optimum solution.

4.2.1. Solution Encoding Scheme

A solution represents an operating procedure. However, the optimizer works on an encoded solution represented as a finite sequence $(A_1, A_2, A_3, ..., A_m)$ where $A_i = (\langle O_i, T_i \rangle, R_i)$. O_i is an operation, T_i denotes the length in time that O_i will be applied and R_i denotes the number of times that the operation O_i with length in time T_i will be repeated.

An operation O_i is defined as an element of the set $O = (I_j, P_{jkl})$ where I_j is a unique integer number that serves as an identifier of the operation, and P_{jkl} is a numeric value of parameter *k* assigned to remotely-operable equipment *l*. A control valve is an example of remotely operable equipment. The implementation is done with three arrays: the first array contains the operation indices; the second array contains the values of length in time and the third array contains the number of times that each operation is repeated.

To simulate a given solution, the solution has to be decoded to an operating procedure described in terms of physical values such as valve positions and time.

4.2.2. Optimization Algorithm

The proposed optimization algorithm is a metaheuristic hybrid algorithm that combines two well-known metaheuristic algorithms: simulated annealing [23] and tabu search [24].

Simulated annealing allows the selection of worse solutions at the early stages of the iterative process in order to avoid local optima. The algorithm reduces the probability of selecting worse solutions, increasingly accepting better solutions. Tabu search stores information of previously evaluated solutions in a memory data structure called tabu list. As a result, tabu search avoids revisiting solutions that have already been evaluated, improving the computational efficiency by avoiding unnecessary simulation runs.

The flow chart of the metaheuristic hybrid algorithm is shown in Figure 5, the following is a detailed explanation of each of the steps in the metaheuristic hybrid algorithm.



Figure 5. Flow diagram of the Hybrid metaheuristic algorithm.

The metaheuristic hybrid algorithm starts by generating a random solution in a codified form making local changes until there is no better solution. The proposed algorithm for the generation of a feasible initial solution that will function as the starting point for the optimization algorithm to find the optimal solution is shown in Figure 6.



Figure 6. Flow diagram of the initial solution generation algorithm.

The initial solution is then fed to the simulator and the plant model is run. The results of the simulation are sent to the optimizer, which evaluates the feasibility of the solution. The feasibility function f(x) determines the difference between the left-hand side and the right-hand side of the constraints. If the generated solution x is feasible, f(x) becomes zero. A solution is considered feasible if it reaches the goal state determined by the problem without violating any of the constraints.

If f(x) > 0, a new solution *y* is generated based on the current solution *x*. Until feasibility is guaranteed, a solution may suffer many changes by the application of multiple neighborhood operators. This is an exploration feature of the algorithm.

Subsequently, the feasibility of solution *y* is evaluated. If f(y) > f(x) solution *x* remains as the current solution, and a new solution *y* is generated, restarting the loop. If f(y) < f(x), solution *y* becomes the new current solution (x = y), and the procedure is repeated. If f(y) equals zero, solution *y* becomes the initial solution and it moves on to the next step.

Afterwards, the algorithm uses this initial feasible solution to produce a neighbor solution in which the local search algorithm explores the space of candidate solutions (the search space) by applying local changes by means of a neighborhood operator (NOP). The NOP can be defined in terms of local rearrangements, such as swapping, moving, or changing one or more elements from the current solution sequence. A neighborhood operator can be applied multiple times to make significant changes in the solution to improve the diversification of solutions through the optimization process.

After selecting and evaluating the neighbor solution, that neighbor solution is added to the tabu list. From this point on, each new solution is tested against the tabu list.

The tabu list contains information of previously evaluated solutions, to avoid searching the same region and avoid repeated simulations. The tabu list works as long-term memory for the algorithm.

If a solution appears in the tabu list, then it will be avoided. When the neighbor solution is feasible, and the objective function of the neighbor solution is better than the objective function of the current solution the neighbor solution will be selected as the current solution. If the total operation time is set as the objective function, a better neighbor solution is the operating procedure that takes less time to reach the goal state.

When the neighbor solution is feasible, and the objective function of the neighbor solution is worse than the objective function of the current solution, then the neighbor solution is accepted as the current solution based on the acceptance probability. The acceptance probability P depends on the values of the objective function of the current and the neighbor solutions, and on a global time-varying parameter T. A typical probability function is a Boltzmann distribution (Equation (4)).

$$P = e^{-\frac{E(y) - E(x)}{T}} \tag{4}$$

where *P* is the probability of selecting a worse neighbor solution, *T* is the temperature of the algorithm, E(y) is the objective function of the neighbor solution, and E(x) the objective function of the current solution.

If the objective function of the neighbor solution is better than the best solution found so far, the neighbor solution becomes the best solution. From this point on, a new current solution is already in place and the iterative process starts again. Throughout the search, better results will be found, while new search areas are evaluated. Once the stopping condition is met, the algorithm delivers the best solution.

5. Case Study

The evaluation of the proposed methodology is carried out by means of a case study on the generation of the optimum operating procedure of a drum boiler. The problem consists of finding an operating procedure that takes the system from a given pressure and steam-flowrate values to the desired pressure and steam mass-flowrate values in the shortest time possible while avoiding excessive thermal stresses in the metal of the wall of the drum boiler. The simulator component was developed using the OpenModelica environment [25,26]. OpenModelica has algorithms for solving differential equation systems, making it possible to observe changes in variables over time. The drum boiler simulation model in OpenModelica is based on the model reported by Rosado et al. [10].

The integration between the OpenModelica simulator and the optimization algorithm was carried out by implementing a two-way interface. First, the interface receives the operating procedure generated by the optimization algorithm. Then, it translates this procedure to a set of parameters for the simulation model. Finally, it evaluates and executes the simulation model. Conversely, the interface receives the results of the simulation and translates them into a format suited to the optimization algorithm.

The optimization problem is formulated based on the work by Belkhir et al. [17]. The goal is to reach given values of pressure and steam outlet flowrate by manipulating the heat inlet valve and the steam outlet valve. Accordingly, the optimization problem is formulated as:

S

$$Min((A*S) + \alpha (P_{sat} - P_{goal})^2 + \beta (q_s - q_{goal})^2)$$
(5)

where

$$=\sum_{t_0}^{t_f} dt \tag{6}$$

Subject to:

$$0 \le V_{pos} \le 1 \tag{7}$$

$$0\,\frac{\mathrm{MW}}{\mathrm{min}} \le \frac{dQ}{dt} \le 25\,\frac{\mathrm{MW}}{\mathrm{min}}\tag{8}$$

$$0 \le Q \le 500 \,\mathrm{MW} \tag{9}$$

$$-10 \,\mathrm{MPa} \le \sigma_D \le 10 \,\mathrm{MPa} \tag{10}$$

Equation (5) seeks to minimize the time it takes for the drum boiler to reach the goal. The parameters α and β from Equation (5) are the ones specified by Belkhir et al. [17]. When A = 0 the problem is reduced to finding a sequence of operations that is feasible but not necessarily optimal [9]. A feasible solution is a solution that reaches the goal state without violating any of the constraints. In this optimization problem P_{goal} is the desired internal pressure, q_{goal} is the desired steam mass-flow rate, α and β are weights, P_{sat} is the steam pressure at the drum boiler, and q_s is the steam flow rate at the drum boiler outlet. Equation (6) represents the time that the system takes to get from the initial state to the final state.

Equation (7) constrains the opening of the steam outlet valve (V_{pos}) to values between 0 (totally closed) and 1 (fully open). Equation (8) ensures that the heat rate does not exceed 25 MW/min. It is a nonlinear constraint because it implies that there could be different heat ramp rates during the process. Equation (9) is a constraint of the accumulated heat limit of the drum boiler which must not exceed 500 MW and Equation (10) is a constraint that avoids excessive thermal stress in the drum boiler that must be less than 10 MPa. The water is supplied by a control system and the steam flow is controlled by a valve. The interaction between the opening of the steam outlet valve and the heat rate in the drum boiler generates steam at pressure P_{sat} which exits at flow rate q_s . The steam can later be sent to a superheater or directly to a steam turbine [9]. For this problem, the goal was set to $P_{goal} = 9$ MPa, $q_{goal} = 180$ kg/s. The weights were set to $\alpha = 10^{-4}$ and $\beta = 10^{-4}$. Parameter A is set to 0 during the initial solution generation and then changed to A = 1 for the rest of the algorithm. The nonlinearity of Equations (7) and (8) add complexity to the problem, which justifies the use of metaheuristic methods to find a solution.

To ensure that the optimization algorithm converges, the stopping condition of the algorithm is met when 1000 iterations are performed. It is worth mentioning that several

tests with more iteration limits were performed, but the algorithm always converged before 1000 iterations.

Each operating procedure is represented according to the encoding scheme explained in Section 4.2.1, in which each operation is formed by combining discrete values of the heat flow and the valve position of the steam outlet valve. This results in the nine operations shown in Table 1.

Table 1. Operations table.

Operation	Heat Flow dQ/dt	V_{pos}
1	8	0.0
2	8	0.6
3	8	1.0
4	16	0.0
5	16	0.6
6	16	1.0
7	24	0.0
8	24	0.6
9	24	1.0

The execution time per operation is set to 60, 120, or 180 s. The repetition parameter is set to vary between 0 to 9. The length of the sequence is fixed to nine elements.

Figure 7 shows an example of an operating procedure. The first element in the sequence represents operation 8 (dQ/dt = 24 MW/min, $V_{pos} = 0.6$) being executed for 60 s and repeated three times.

Operation	8	9	4	5	4	8	7	2	8
Time	60	120	180	60	180	120	120	120	60
Repetition	3	2	8	9	3	9	6	4	2

Figure 7. Solution representation of an operating procedure for the drum boiler optimization problem.

The feasibility function f(x) is calculated with Equation (11). An extra penalty can be applied to f(x) in case the total time of the generated sequence is less than 1200 s, a value too low to be feasible.

$$if\{(t < 1200, f(x) = V_1 + 100; t > 1200, f(x) = V_1)\}$$

$$V_1 = (400 - Q) + f_s + G$$

$$G = \alpha (P_{sat} - P_{goal})^2 + \beta (q_s - q_{goal})^2$$
(11)

where Q is the accumulated heat that must reach 400 MW, f_s represents the number of times the thermal stress exceeded the 10 MPa limit throughout the process and G represents how far is the sequence of approaching the steam pressure and steam outflow goal.

During the optimization process, new solutions are generated using the NOP. The NOP takes an existing solution and makes a mutation by randomly change one of the sequence elements (operation, time, and repetition). A neighborhood operator can be applied multiple times as a solution-diversification strategy. Figure 8 shows an example of NOP applied four times.

ACTUAL SOLUTION

3	4	6	3	6	6	7	4	3		
60	120	180	180	60	60	60	180	120		
4	2	1	5	5	7	3	7	3		
ONE MUTATION										
3	8	6	3	6	6	7	4	3		
60	120	180	180	120	60	60	180	120		
4	2	6	5	5	7	3	7	3		
TWO N	TWO MUTATIONS									
3	8	6	3	6	6	2	4	3		
60	120	180	180	120	180	60	180	120		
4	2	6	5	5	7	3	7	0		
THREE	E MUTA	TIONS				-		-		
3	8	6	5	6	6	2	4	3		
60	60	180	180	120	180	60	180	120		
4	8	6	5	5	7	3	7	0		
FOUR MUTATIONS										
3	8	6	5	6	1	2	4	3		
60	60	180	180	120	180	60	120	120		
4	8	6	5	5	1	3	7	0		

Figure 8. Mutation process from one to four mutations.

The number of times that NOP is applied is set based on the value of f(x) as shown in Table 2. The number of times NOP is applied depends on the length of the solution for the given problem. In this case study, the maximum number of mutations is four because it changes half the values of the previous solution in the worst scenario for f(x).

Table 2. Number of mutations according to the feasibility function of the operating procedure.

f(x)	Number of Mutations
>400	4
>300	3
>50	2
<50	1
≈ 0	0

For the metaheuristic hybrid algorithm and the simulated annealing algorithm, it is necessary to specify the acceptance probability function. This function is represented by Equation (12):

$$P = e^{-\frac{((t(y) - t(x)) * \gamma)}{T}}$$
(12)

where *P* is the probability of selecting a worse neighbor solution, *T* is a parameter that gradually decreases as the algorithm proceeds (*T* is also known as the annealing temperature), t(y) is the final time of the neighbor solution and t(x) the final time of the actual solution because the time is the value that is sought to optimize in this problem. γ is a parameter that magnifies the difference between two solutions.

To select the value of γ , the value of the Boltzman distribution was analyzed with a difference of 50 s between two solutions, varying the value of γ between 1 and 15. The value of $\gamma = 10$ was selected as it starts with a probability slightly larger than 50% but steadily decreasing through the iterations. Figure 9 shows the probability of selecting a worse solution (50 s worse) using different γ values in a run of 1000 iterations.



• Better solution • worse solution, Y=1 • worse solution, Y=5 • worse solution, Y=10 • worse solution, Y=15

Figure 9. Probability of selecting a better or a worse solution (50 s worse), using different γ values, depending on the progress of a run of 1000 iterations.

6. Experiments and Results

To evaluate the proposed approach, two experiments were carried out:

- Experiment A, which uses a randomly generated operating procedure as the initial solution for the metaheuristic hybrid algorithm. The results were then compared against a benchmark solution [17] and a solution obtained with the micro genetic algorithm described in [10].
- Experiment B, which compares the metaheuristic hybrid algorithm, the simulated annealing algorithm, and the tabu search algorithm. The comparison considers two different initial solutions: a randomly generated solution; and a feasible solution generated by the procedure explained in Figure 6.

6.1. Experiment A

The benchmark solution is a representative startup profile reported by Belkhir et al. [17]. In this solution, the operating procedure consists of keeping the heat input at a constant value of 8 MW/min and the steam output valve fully open since the very beginning. After executing the operating procedure with the OpenModelica simulation model, the startup completed in 3000 s.

Figure 10 shows the plot of function $f(P_{sat}, q_s) = \alpha (P_{steam}(t) - P_{goal})^2 + \beta (q_{steam} - q_{goal})^2$ which measures the distance to the goal state over time.

The Belkhir operating procedure was a feasible solution as it never exceeded the limits imposed by the thermal stress constraint, maintaining a stress value between -10 MPa and 10 MPa. The goal state was reached after 3000 s. The stress profile is shown in Figure 11.

The micro genetic algorithm (mGA) implemented the same encoding scheme of this paper. The probabilities used in all the experiments were 10% for mutation and 20% for crossover. The population of the mGA consisted of 5 individuals, and the termination criteria were set to a maximum of 40 generations and 20 epochs, respectively.

The metaheuristic hybrid algorithm was initiated with a randomly generated solution and stopped after 1000 iterations. Table 3 shows the solution obtained as a result of the optimization. Table 4 shows the decoded operating procedure based on the operations shown in Table 1. After reaching the desired goal, the heat inlet valve and the steam outlet valve took the values of 0 MW/min and full open respectively.



Figure 10. Change of the distance to the goal state over time with the benchmark solution.



Figure 11. Changes of thermal stress in the drum boiler over time with the benchmark solution.

Table 3. Best solution obtained with the metaheuristic hybrid algorithm and a randomly generated initial-solution.

Operation	8	3	9	9	7	2	8	1	3
Time	60	120	180	120	120	60	120	120	60
Repetitions	1	7	8	3	8	5	1	9	5

Table 4. Operating procedure decoded from the solution shown in Table 3.

Time	Heat Inlet Valve	Steam Outlet Valve	Accumulated Time
60	24	0.6	60
840	8	1	900
660	24	1	1560
-	0	1	-

With this operation procedure, the drum boiler arrived at the goal state in 1560 s. Figure 12 shows the change of the distance to the goal state over time over time for the metaheuristic hybrid algorithm, the benchmark solution, and the micro genetic algorithm.



Figure 12. Change of the distance to the goal state over time over time for the benchmark solution (red line), the micro genetic algorithm (black line) and the proposed metaheuristic hybrid algorithm (green line).

The operating procedure obtained with the metaheuristic hybrid algorithm produced a feasible thermal stress profile of the thick-walled component. This thermal stress profile had a similar shape pattern and magnitude as the one obtained with either the benchmark solution or the solution obtained with the micro genetic algorithm, which means that the drum boiler integrity was not affected. Figure 13 shows the behavior of the thermal stress, which is related to the structural integrity of the drum boiler.



Figure 13. Comparison of the the thick-walled thermal stresses generated with the benchmark solution (red line), the micro genetic algorithm solution (black line) and the metaheuristic hybrid algorithm solution (green line).

The feed water flow had to be controlled so that the water level inside the drum was kept at its set point. A PI controller was used for this purpose. Figure 14 shows the behavior of the water level during the entire drum boiler startup process, where the metaheuristic hybrid algorithm achieved a more stable pattern compared to the micro genetic algorithm and the benchmark solution.



Figure 14. Changes of water level obtained with the benchmark solution (red line), the micro genetic algorithm solution (black line) and the metaheuristic hybrid algorithm solution (green line).

Figure 14 shows that the water level changed abruptly during the drum boiler startup process. The instability of the liquid level could be reduced with the use of a non-linear controller that can adapt to the non-linearity of the model or the use of the gain scheduling approach which involved the application of different controller tuning parameters as a process transitions from one operating range to another [27]. However, the magnitude of the changes in water level was considered tolerable for this experiment.

The best result was obtained at the 654th iteration. Due to the memory strategy, the simulation of 396 previously simulated solutions was avoided. The experiment took 216 min on a computer with 4.00 GHz Intel Xeon W-2125 CPU and 32 GB of RAM, running Windows 10 Pro. In summary, the proposed approach could synthesize a startup operating procedure that reached the goal state in 48% less time than the benchmark solution, without sacrificing feasibility.

6.2. Experiment B

Experiment B compares the proposed metaheuristic hybrid algorithm against two well-known metaheuristic algorithms: simulated annealing and tabu search.

To prove if the difference in the startup time of each algorithm is significant, a hypothesis test was conducted. To do this test, experiments were carried out for a randomly generated initial solution and a feasible solution that was generated according to the procedure shown in Figure 6. A total of 10 experiments were performed for each algorithm as shown in Table 5. In either algorithm, each experiment was run with 1000 iterations as stopping condition, which was shown to guarantee convergence in all cases.

Table 5. Experiment design for experiment B.

Initial Solution	Hybrid Algorithm	Simulated Annealing	Tabu Search
Randomly Solution	10 experiments	10 experiments	10 experiments
Feasible Solution	10 experiments	10 experiments	10 experiments

The test aimed at proving whether there was a significant difference in the average startup time obtained by each of the algorithms. To prove that, we used the t-test and the mean start up time of 10 experiments. The start-up time means were considered significantly different when $p \le 0.05$. Conversely, start-up times means were not considered significantly different if p > 0.05. In all the experiments, the result was p < 0.01, indicating that in all the experiments the start-up times means were considered significantly different.

For the time to complete the start-up, the following hypothesis test was carried out, which sought to verify or refute if there was a significant difference between the solution generated by the metaheuristic hybrid algorithm and the tabu search algorithm; and between the metaheuristic hybrid algorithm and the simulated annealing algorithm:

$$h_0: \mu_x - \mu_y = 0 \text{ versus } h_1: \mu_x - \mu_y \neq 0$$
 (13)

where μ_x is the average of the startup time obtained with the metaheuristic hybrid algorithm and μ_y is the average of the startup time obtained with either simulated annealing or tabu search.

Tables 6 and 7 show the mean and standard deviation of the 10 experiments for each method. Table 8 shows the results obtained with the application of the *t*-test in experiment B.

Table 6. Comparison of mean and standard deviation values with a randomly generated solution as an initial solution.

	Hyl Algo	brid rithm	Tabu Search		Simmulated Annealing		Micro Genetic Algorithm	
	Best	CPU	Best	CPU	Best	CPU	Best	CPU
Mean	1626	181	1606	607.2	1636	313.5	1800	156
Stand. Dev.	59.67	37.76	59.67	137.92	60.96	5.17	60.01	39.48

	Hyl Algo	brid rithm	rid Tabu ithm Search		Simmulated Annealing		Micro Genetic Algorithm	
	Best	CPU	Best	CPU	Best	CPU	Best	CPU
Mean	1608	227.2	1590	640.5	1598	318.9	1800	160
Stand. Dev.	41.31	21.81	45.46	88.41	50.29	0.32	30.98	23.01

Table 7. Comparison of mean and standard deviation values with a feasible solution as an initial solution.

In Tables 6 and 7, the "best" result was the best start-up time measured in seconds and the "CPU" result was the required time for the CPU to achieve it measured in minutes.

Table 8. *p*-values for the comparison of the minimum startup time of the metaheuristic Hybrid Algorithm (HA) vs. Tabu Search (TS), Simmulated Annealing (SA) and the micro Genetic Algorithm (mGA).

Initial Solution	HA vs. TS	HA vs. SA	mGA
Randomly generated solution	0.463	0.715	0.0001
Feasible solution	0.374	0.633	0.0001

For the case of minimum drum boiler startup time, the *t*-test shows that there was no significant difference in the result obtained with the metaheuristic hybrid algorithm compared to the result obtained with the tabu search algorithm and simulated annealing. However, this test shows that there was a significant difference compared to the micro genetic algorithm.

In the case of computational time, as shown in the box plots of Figure 15, the results indicate that the computational time of the metaheuristic hybrid algorithm was better than the computational time of either the tabu search algorithm, simulated annealing, and micro genetic algorithm. As shown in Figure 16 the computational time of the metaheuristic hybrid algorithm was better than the computational time of both simulated annealing

and tabu search algorithm, but it was not better than the computational time of the micro genetic algorithm. However, in any case, the solutions proposed by the micro genetic algorithm were better than the proposed by the metaheuristic hybrid algorithm in terms of the start-up time.



Figure 15. Box plot comparing the computation time for all four algorithms using a randomly generated solution as the initial solution.



Figure 16. Box plot comparing the computation time for all four algorithms using a feasible solution as the initial solution.

7. Conclusions and Future Work

This paper presents an approach for the synthesis of the operating procedures of a plant system. Specifically, a metaheuristic optimization algorithm was developed that combines two characteristics of other metaheuristic algorithms, namely the cooling element from the simulated annealing algorithm and the memory structure of the tabu search algorithm.

From the results of the experiments, it can be concluded that the proposed methodology can synthesize an operating procedure for the startup of a drum boiler of a thermal power plant that takes 48% less time to reach its goal state than a representative startup profile found in the literature.

From the results of the experiments, it is evident that the metaheuristic hybrid algorithm performs better than the individual algorithms in terms of computational time. However, results from the t-test indicate that there is no significant difference in the drum boiler startup time result using the metaheuristic hybrid algorithm against the simulated annealing or tabu search algorithms. Despite the advantage in computational time, this result of the metaheuristic hybrid algorithm can still be improved by modifying the exploration strategy of the hybrid algorithm.

Additionally, the results reveal that the metaheuristic hybrid algorithm can find a better solution than the one found with the micro genetic algorithm. The hybrid algorithm tends to find better solutions by cooling element and the tabu list. The "cooling" element allows the selection of new "worse" solutions at the early stages of the iterative process in order to avoid local optima. Then the algorithm "cools" as it converges, so the probability of selecting "worse" solutions decreases, accepting only better solutions. On the other hand, the tabu list is a memory structure that stores information of previously evaluated solutions. As a result, the algorithm avoids visiting again solutions that have already been evaluated, improving the computational efficiency by avoiding unnecessary simulation runs.

As future work, different optimization algorithms can be studied while keeping the same simulation model. On the other hand, more complex simulation models for the drum boiler can be used to increase fidelity in the results or eliminate disturbances in the model caused by the controller. Finally, the proposed approach for the synthesis of operating procedures could be used in processes other than the drum boiler startup.

Author Contributions: All the authors contributed to this research. Conceptualization, E.G., E.R.-T. and M.A.Z.-G.; methodology, E.G., E.R.-T., M.A.Z.-G. and R.B.; software, E.G., E.R.-T. and M.A.Z.-G.; investigation E.G. and E.R.-T.; formal analysis, E.G. and R.B.; writing original draft preparation, E.G. and E.R.-T.; review and editing, M.A.Z.-G. and R.B.; project administration and funding acquisition, R.B. All authors read and agreed to the published version of the manuscript.

Funding: National Council of Science and Technology (CONACYT).

Institutional Review Board Statement: Not Applicable.

Data Availability Statement: Not Applicable.

Acknowledgments: Thanks to the School of Engineering and Sciences of the Tecnologico de Monterrey, for the support provided to carry out this work and the fund of the National Council of Science and Technology (CONACYT) to carry out this research.

Conflicts of Interest: The authors declare no conflict of interest.

References

- 1. Ulbig, A.; Andersson, G. Analyzing operational flexibility of electric power systems. *Int. J. Electr. Power Energy Syst.* 2015, 72, 155–164. [CrossRef]
- Kintner-Meyer, M.C.; Homer, J.S.; Balducci, P.J.; Weimar, M.R. Valuation of Electric Power System Services and Technologies; Technical Report; Pacific Northwest National Lab. (PNNL): Richland, WA, USA, 2017.
- 3. Alizadeh, M.; Moghaddam, M.P.; Amjady, N.; Siano, P.; Sheikh-El-Eslami, M. Flexibility in future power systems with high renewable penetration: A review. *Renew. Sustain. Energy Rev.* 2016, 57, 1186–1193. [CrossRef]
- Taibi, E.; Nikolakakis, T.; Gutierrez, L.; Fernandez, C.; Kiviluoma, J.; Rissanen, S.; Lindroos, T.J. Part 1, overview for policy makers. In *Power System Flexibility for the Energy Transition*; International Renewable Energy Agency: Abu Dhabi, United Arab Emirates, 2018.
- Dong-Mei, J.; Jia-Qi, S.; Quan, S.; Heng-Chao, G.; Jian-Xing, R.; Quan-Jun, Z. Optimization of start-up scheduling and life assessment for a steam turbine. *Energy* 2018, 160, 19–32. [CrossRef]
- 6. Liu, Z.; Karimi, I. Simulation and optimization of a combined cycle gas turbine power plant for part-load operation. *Chem. Eng. Res. Des.* **2018**, *131*, 29–40. [CrossRef]
- Rossi, I.; Sorce, A.; Traverso, A. Gas turbine combined cycle start-up and stress evaluation: A simplified dynamic approach. *Appl. Energy* 2017, 190, 880–890. [CrossRef]
- 8. Anisimov, A.; Klub, M.; Sargsyan, K.; Eritsyan, S.K.; Petrosyan, G.; Avtandilyan, A.; Gevorgyan, A. Optimization of Start-Up of a Fully Fired Combined-Cycle Plant with GT13E2 Gas Turbine. *Power Technol. Eng.* **2016**, *49*, 359–364. [CrossRef]
- 9. Åström, K.J.; Bell, R.D. Drum-boiler dynamics. Automatica 2000, 36, 363–378. [CrossRef]
- 10. Rosado-Tamariz, E.; Zuniga-Garcia, M.A.; Campos-Amezcua, A.; Batres, R. A framework for the synthesis of optimum operating profiles based on dynamic simulation and a micro genetic algorithm. *Energies* **2020**, *13*, 677. [CrossRef]
- 11. Batres, R. Generation of operating procedures for a mixing tank with a micro genetic algorithm. *Comput. Chem. Eng.* 2013, 57, 112–121. [CrossRef]
- 12. Batres, R.; Soutter, J.; Asprey, S.P.; Chung, P. Operating procedure synthesis: Science or art? *Knowl. Eng. Rev.* 2002, 17, 261. [CrossRef]

- 13. Adam, E.; Marchetti, J. Dynamic simulation of large boilers with natural recirculation. *Comput. Chem. Eng.* **1999**, 23, 1031–1040. [CrossRef]
- Åkesson, J.; Årzén, K.E.; Gäfvert, M.; Bergdahl, T.; Tummescheit, H. Modeling and optimization with Optimica and JModelica. org—Languages and tools for solving large-scale dynamic optimization problems. *Comput. Chem. Eng.* 2010, 34, 1737–1749. [CrossRef]
- Franke, R.; Vogelbacher, L. Nonlinear model predictive control for cost optimal startup of steam power plants (nichtlineare modellprädiktive regelung zum kostenoptimalen anfahren von dampfkraftwerken). *at-Automatisierungstechnik* 2006, 54, 630–637. [CrossRef]
- 16. Krüger, K.; Franke, R.; Rode, M. Optimization of boiler start-up using a nonlinear boiler model and hard constraints. *Energy* **2004**, 29, 2239–2251. [CrossRef]
- 17. Belkhir, F.; Cabo, D.K.; Feigner, F.; Frey, G. Optimal startup control of a steam power plant using the JModelica platform. *IFAC-PapersOnLine* **2015**, *48*, 204–209. [CrossRef]
- Franke, R.; Rode, M.; Krüger, K. On-line optimization of drum boiler startup. In Proceedings of the 3rd International Modelica Conference, Linköping, Sweden, 3–4 November 2003; Volume 3.
- 19. Zhang, T.; Zhao, Z.; Li, Y.; Zhu, X. The simulation of start-up of natural circulation boiler based on the Astrom-Bell model. *AIP Conf. Proc.* **2017**. *1794*, 040003.
- 20. Yu, T.; Chan, K.W.; Tong, J.; Zhou, B.; Li, D. Coordinated robust nonlinear boiler-turbine-generator control systems via approximate dynamic feedback linearization. *J. Process Control* **2010**, *20*, 365–374. [CrossRef]
- 21. Cannon, G. Process Technology Equipment; Pearson Education, Inc.: New York, NY, USA, 2019.
- 22. Liu, S.; Zhao, B.; Zhao, S.; Zhang, L.; Wu, L. An Intelligent Bio-Inspired Cooperative Decoupling Control Strategy for the Marine Boiler-Turbine System with a Novel Energy Dynamic Model. *Energies* **2019**, *12*, 4659. [CrossRef]
- Kirkpatrick, S.; Gelatt, C.D.; Vecchi, M.P. Optimization by simulated annealing. *Science* 1983, 220, 671–680. [CrossRef] [PubMed]
 Glover, F.; Laguna, M.; Marti, R. Principles of tabu search. *Approx. Algorithms Metaheuristics* 2007, 23, 1–12.
- 25. Fritzson, P.; Pop, A.; Asghar, A.; Bachmann, B.; Braun, W.; Braun, R.; Buffoni, L.; Casella, F.; Castro, R.; Danós, A.; et al. The OpenModelica integrated modeling, simulation and optimization environment. In Proceedings of the 1st American Modelica Conference, Cambridge, MA, USA, 9–10 October 2018; Modelica Association: Linköping, Sweden, 2018; pp. 8–10.
- 26. Casella, F.; Leva, A. Object-oriented modelling & simulation of power plants with modelica. In Proceedings of the 44th IEEE Conference on Decision and Control, Seville, Spain, 15 December 2005; IEEE: Piscataway, NJ, USA, 2005; pp. 7597–7602.
- 27. Mercader, P.; Cánovas, C.; Baños, A. Control PID multivariable de una caldera de vapor. *Rev. Iberoam. Autom. Inform. Ind.* 2019, 16, 15–25. [CrossRef]