DOI: 0

CompAMa

www.compama.co.usb.ve

Bull. Comput. Appl. Math., Vol.5, No.1, 2017

ISSN 2244-8659

Fitting of parameters for a temperature control model by means of continuous derivative-free optimization: a case study in a broiler house

Bruno H. Cervelin [1], *Dante Conti* [2], *Denise T. Detsch* [3], *Maria A. Diniz-Ehrhardt* [4], *José Mario Martínez* [5]

CompAMa Vol.5, No.1, pp.117-139, 2017 - Accepted July 11, 2017

Abstract

Intensive broiler production requires of accurate control systems aimed to maintain ideal conditions inside the facilities. The achievement of an appropriate environment guarantees good performance and sustainability of the production. Control and monitoring of temperature is a key factor during the production cycle. In countries with tropical and subtropical climate, such as Brazil, high values of temperatures can affect negatively the broiler production. Based on a temperature control model developed by the authors, this research is focused on the determination and fitting of the intrinsic parameters of the model. Consecutive executions

[1] Department of Applied Mathematics, Institute of Mathematics, Statistics and Scientific Computing, University of Campinas, SP, Brazil (bcervelin@gmail.com).

[2] Department of Applied Mathematics, Institute of Mathematics, Statistics and Scientific Computing, University of Campinas, SP, Brazil (conti@ime.unicamp.br).

[3] Department of Engineering and Exact Sciences, Federal University of Paraná (Palotina Sector), Brazil (denise.detsch@ufpr.br).

[4] Department of Applied Mathematics, Institute of Mathematics, Statistics and Scientific Computing, University of Campinas, SP, Brazil (cheti@ime.unicamp.br).

[5] Department of Applied Mathematics, Institute of Mathematics, Statistics and Scientific Computing, University of Campinas, SP, Brazil (martinez@ime.unicamp.br).

of the model and changes in the facilities suggest adapting parameters constantly under the perspective of real-time systems. Four strategies of derivative-free optimization were applied to adjust the parameters of the model. Experiments were conducted with data collected from a pilot farm in South-eastern Brazil. Results demonstrated that the process of updating parameters needs to be implemented on the temperature control model. BOBYQA method resulted to be the best strategy to be taken into consideration for the improvement of the system.

Keywords: Parameter optimization, broiler production, control of temperature, derivative-free methods, real-time.

1 Introduction

Livestock production requires control, monitoring and surveillance of operating conditions in order to guarantee good performance, productivity, sustainability and animal comfort. Basically, these operating conditions are related to thermal variables: temperature, relative humidity, air velocity/wind speed amongst other factors and their interaction with automated or semi-automated devices inside the facilities (ventilation systems and controllers). Temperature is one of the key variables to be kept under control during a rearing process [1]. For intensive broiler production (chicken meat), temperature inside the facilities (broiler houses) is a crucial factor that needs to be controlled and monitored almost in real-time [2, 3].

Brazil is one of the top three broiler producers in the world together with United States of America and China. In countries where climate is tropical and subtropical, such as Brazil, variations of temperature affects the rearing process by putting at risk the thermal comfort of the animals. Discomfort produces heat stress and high rates of mortality [4]; in consequence, production is affected negatively in terms of weight conversion, feed efficiency and animal welfare [5]. Therefore, an efficient control of thermal conditions is necessary to maximize the production and guarantee its sustainability.

An efficient control of thermal conditions is commonly supported by ventilation systems and controllers inside the facilities. The achievement of ideal conditions is associated to the interaction with automated controllers and operating policies which respond to the current conditions (thermal variables) by switching on or off devices (exhaust fans, cooling pads and humidifiers). The challenge is to guarantee the most comfortable microclimate inside the broiler houses. Keeping a good microclimate is a complex problem. Some approaches deal with quantitative methods in data-driven models which use Computation Fluid Dynamics (CFD), Statistics, Data Mining, Artificial Intelligence and Applied Mathematics in order to understand, control and support accurately the thermal conditions at the facilities by interacting with automated devices. All these approaches are called as precision livestock farming (PLF). Some interesting researches can be found in [6–10].

Recently, authors of this research developed a temperature control model aimed to support thermal conditions inside broiler houses. The model combines applied mathematics, optimization and some empirical considerations in order to equilibrate accuracy with fast execution for real-time practice. The developed model, described in details in [11], uses a one-dimensional representation of the broiler house, with left and right walls, in which the temperature is propagated by a diffusion process, subject to boundary conditions given by external temperature and initial conditions provided by several sensors placed along the house. The ventilation system (Exhaust fans) is modeled as heat sources that contribute to balance the temperature inside the facilities. This model would require of the determination/ fitting of three parameters associated to it for its correct use as a supporting tool. The fitting process is necessary when changes or structural modifications of the broiler house are evidenced. These changes and the real-time approach could produce significant variations of the coefficients, so the update of parameters becomes a key task to be implemented.

In this research, the determination and fitting of the parameters associated to the temperature control model [11] is managed by using continuous derivative-free optimization. This family of optimization methods tries to achieve the optimal value of an objective function without evaluating or approximating its derivatives. They are used, for example, when a black-box objective function is present, i.e., when the actual equation of the objective function is not available, so its derivatives are also not available. Parameter optimization problems usually fit in this family of problems.

Some related problems are cited in Audet and Orban [12]. They proposed an objective function to optimize the parameters of a trust region method in terms of the processing time of the method. Cervelin [13] proposed some variations of the objective function and optimized the parameters of a derivative-free method in relation to the number of function evaluation performed by it. These two works focus more in the optimization process than the application of the techniques in real-world problems. Wild [14] used derivative-free techniques to solve the parameter estimation of problems related to nuclear physics, and Mukherjee [15] enunciates some derivative-free techniques that are implemented for metal cutting processes.

Under these premises, the paper is aimed to describe how continuous derivative-free optimization is applied to the temperature control model. Parameters associated to the model correspond to the diffusion coefficient inside the house, the diffusion coefficient in the walls and the effect of each exhaust fan on the variation of the temperature in one unit of time. Fitting these coefficients to actual data is a task that may be accomplished by minimization algorithms. The temperature model is part of a control model and, since it can be modified in order to improve the control model efficiency, the merit function that measures the quality of the approximation can change. So, even if it is not impossible to compute the derivatives of the merit function, the necessity of changing the model structure led us to the use of derivative-free methods. These methods allow taking advantage of enough flexibility and satisfactory speed of execution for real-time situations.

The structure of the paper is presented as follows: Section 2 describes a summarization of the temperature control model obtained from [11] together with the control process and its relation with derivative-free techniques, Section 3 gives a basic description with flowcharts of the continuous derivative-free optimization techniques used in the research. In Section 4 the case study for a Brazilian broiler house is detailed with its corresponding results and discussion. Finally, the paper is closed in Section 5 with the conclusions.

2 The temperature control model

2.1 Basics of the model

The Broiler House is represented as a segment $[0, L]$, where L represents the length of the house, the segments $[-a, 0]$ and $[L, L+a]$ represent the left-wall and the right-wall respectively. Thus, a may be thought as the thickness of each wall. The control devices have the property of decreasing the internal temperature (T) u Celsius degrees per time unit, where $u = u(x,t)$ is a function that depends on the control decisions. So, the Partial Differential Equation (PDE) problem is given by:

$$\frac{\partial T}{\partial t}(x,t) = p_2 \frac{\partial^2 T}{\partial x^2}(x,t) \text{ if } x \in [-a, 0],$$

$$\frac{\partial T}{\partial t}(x,t) = p_1 \frac{\partial^2 T}{\partial x^2}(x,t) - u(x,t) \text{ if } x \in [0,L],$$
$$\frac{\partial T}{\partial t}(x,t) = p_2 \frac{\partial^2 T}{\partial x^2}(x,t) \text{ if } x \in [L,L+a],$$
$$T(x,0) \text{ given for all } x \in [-a,L+a],$$
$$T(-a,t) = T(L+a,t) \text{ given for all } t \geq 0,$$
$$\frac{\partial^2 T}{\partial x^2}(0,t) = \frac{\partial^2 T}{\partial x^2}(L,t) = 0 \text{ for all } t \geq 0.$$

The control function will be assumed to depend on the control devices (here, the ventilation system or exhaust fans). Moreover, it is assumed that the control devices have a finite number of possible states $d_0, d_1, \ldots, d_N$. For example, d_j indicates that the number of connected exhaust fans is j. To each possible state of the controls d_j a function $u_{d_j}(x,t)$ is associated and defined as follows:

$$u_{d_j}(x,t) = \alpha - 0.05 j p_3.$$

So, in the absence of connected exhaust fans, the internal temperature increases α degrees per time unit but the activation of each fan decreases the temperature $0.05p_3$ degrees per time unit.

The one-dimensional PDE model described has three parameters that need to be fitted to real data before (or during) the execution in broiler houses. As mentioned before, the three parameters correspond to the diffusion coefficient inside the house, the diffusion coefficient in the walls and the effect of each exhaust fan on the variation of the temperature in one unit of time. Fitting of these parameters is performed by derivative-free algorithms.

2.2 The Control Process and Fitting of Parameters

The PDE described in Section 2.1 predicts the temperature inside the broiler house using predictions of the external one. To do so, the PDE uses some internal parameters (diffusion coefficients and fans effects). These parameters were previously estimated for a particular broiler house, but, since they depend on several other parameters that are not considered in the model (quantity and size of the birds, broiler house wall material, the maintenance of the exhaust fans and other specific parameters of the house), they should be adjusted for each broiler house. And more, since some of the non-considered parameters are time dependent, it is reasonable to recalibrate the parameters during the simulation. The parameters are updated if the predicted internal temperature is very different from the observed one.

During the rearing process, a set of sensors are strategically positioned inside the broiler house. These sensors provide the temperature in real time. Human operator or an automated operator (controller) reads these current temperatures. In addition, an external sensor collects the outside temperature and its forecasted values for a next period. These values are also transmitted and read by the operator. According to the values of the registered temperatures the operator decides to switch on or off the correct number of exhaust fans in order to approximate the internal temperature to the ideal one which depends basically on the infrastructure of the facility, age and breed of the animal. For that purpose, the operator forecasts the internal temperature for a period of (say) one hour and, based on this prediction, decides how to proceed with the exhaust fans.

To make the decision, the operator compares the results of connecting or disconnecting every combination of the exhaust fans along the period under consideration and chooses the combination that, according to the prediction, produces the best profile of internal temperatures in terms of animal comfort.

The process of choosing the configuration of connected fans is a combinatorial optimization problem where a comfort function, represented by the difference between the achieved temperature and the ideal one is optimized. In turn, the evaluation of the comfort-like function involves the experience of the operator (if human), the consolidated advice in some operation sheet (also reflecting human experience) or the execution of a prediction model, in the case that the control is automatic.

In the temperature control model, the prediction is given by the solution of the one-dimensional PDE system briefly described in Section 2.1. As said above, the parameters of the PDE may be modified during the process, and for applicability of the temperature control model, it must be done in real-time.

In fact, the whole system involves permanent collection of data and parallel fitting of the parameters to updated data. This self-correcting scheme should improve the success of the operation as far as time goes on in a single broiler house. However, structural modifications of the broiler house may produce significant variations of the fitted coefficients. Therefore, it is important to implement efficient and reasonably fast fitting algorithms to the parameters of the temperature control model.

In consequence, the main goal is to minimize a measure function m. This function gives the performance of the method. In this specific case, it will be represented by the 2-norm of the difference of the predicted temperature and the observed one at several instants of times. Usually this kind of problem has a black-box objective function, i.e., the equation that defines the objective function is not known. Since there is no access to its expression, then there is no access to its derivatives, so derivative-free methods can be used to optimize the function.

It also supposed that function m depends on some parameters p of the method and the set of problems used to find the optimal parameters is called training set. In this research, the measure function m is the error performed by the first order PDE when solving the problems on the training set. The main parameters of the PDE model are the diffusion of temperature in the broiler house (p_1), the diffusion temperature in the walls (p_2), and the effect of each exhaust fan on the variation of temperature in one time unit (p_3). For instance, if the predicted variation of temperature according to the diffusion model at the position x from time t to $t+1$ in the absence of exhaust fans is α, the predicted variation with one connected exhaust fan will be $\alpha - 0.05p_3$.

The optimization problem which tries to find the values of parameters that best fits the data is:

$$\begin{array}{ll} \text{minimize} & m(p) \equiv \dfrac{\|T_S(p) - T_r\|_2^2}{n} \\ \text{such that} & 0 \leq p_i \leq 40 \quad i = 1, 2, 3, \end{array} \tag{1}$$

where n is the number of simulated temperatures, p is the vector with the parameters in which we are interested, $T_S(p)$ is a vector with simulated temperatures for different instants of time using the parameters p and T_r is a vector with the observed temperatures at the same times. Note that the upper limit 40 for p_3 means that each exhaust fan decreases the temperature 2 degrees Celsius per unit of time.

Three main strategies were selected to solve the problem: the first one uses BOBYQA [16–18] that optimizes the objective function using a trust region model based on quadratic interpolation; the second one uses Pattern Search [19] which tries to find the optimal point moving through a positive generating set of directions, and finally the third strategy uses SID-PSM [20] which combines the Pattern Search approach with trust regions based on simplex derivatives. These strategies are able to deal with multidimensional derivative-free problems, as it happens in this research. An optional (the fourth one) strategy was also included which deals with Golden Section

Search [21]. Although, this last method is applicable only for one-dimensional problems, it was used by optimizing the parameters in a sequential way, i.e., each parameter at a time.

Results found by each of the four strategies will help to determine the final decision about the type of minimization method that should be used in practical situations from now on. This decision is made by establishing a comparison/benchmarking process within the whole set of strategies.

3 Basic background of continuous derivative-free optimization techniques

3.1 BOBYQA

BOBYQA (Bound Optimization BY Quadratic Approximation) was presented in [18]. Some theoretical properties are described in [16, 17]. As its name states, it tries to optimize a problem approximating the objective function by quadratic models, which are built using interpolation points. These points are the ones where the objective function was already evaluated at previous iterations.

In each iteration we build a model, optimize it in a trust region and verify if the minimizer of the model decreases the value of the objective function. If it happens, we accept this point as the new approximation to the minimizer and update the points of the interpolation and the trust region size; otherwise, we update the points of the interpolation and decrease the trust region size.

Figure 1 shows the BOBYQA process.

3.2 Pattern search

Pattern Search [19] is a derivative-free optimization method that tries to find the optimal point of an objective function f moving through some fixed directions of a set D.

D must be a positive spanning set, i.e., any vector of the work space can be described as a positive linear combination of the directions in D.

For each iteration, there is an approximation x_k of the minimizer and it is updated by evaluating $f(x_k + \alpha d_i)$ where d_i is a vector of D and α is the step parameter of the method. If this function value is less than $f(x_k)$, then set

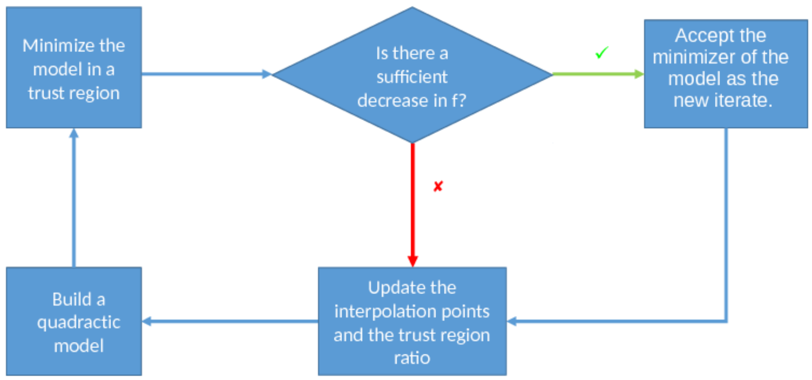


Fig. 1: Flowchart of the BOBYQA method.

$x_{k+1} = x_k + \alpha d_i$ and it is possible to update the step parameter increasing its size, otherwise, another direction in D should be used. If all the directions were used and there is no improvement in the function value, then the step parameter sizeshould be decreased.

Usually, the stop criteria of this method is the step parameter, if it is smaller than a value at iteration k, then x_k is an approximation of the minimizer of f.

Figure 2 shows a flowchart describing the method.

3.3 SID-PSM

The SID-PSM (Simplex Derivatives in Pattern Search Method) method was presented at [20]. It is a combination of Pattern Search with trust region method.

In each iteration the points in which the function is evaluated are stored. Then, given an approximation x_k for the minimizer of f and a positive spanning set D, the method can be described in three steps:

1. Look for a subset of the stored points with a good geometry to approx-

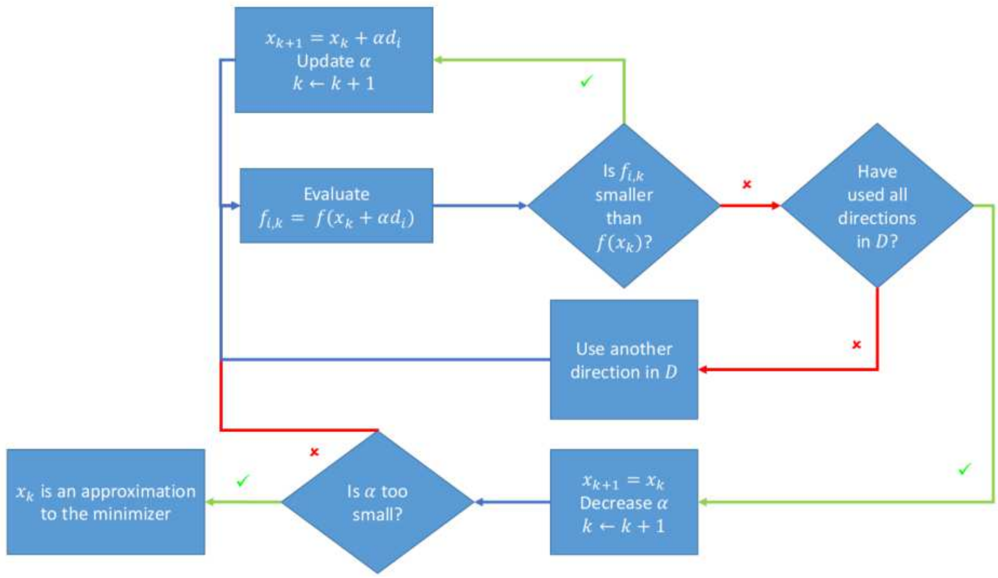


Fig. 2: Flowchart of the pattern search method.

imate the objective function;

2. If there is a set of points with good geometry, we build a model m, and minimize it. If its minimizer decreases the objective function value, we accept this point as the new approximation for the minimizer of f. Else, go to step 3.

3. In the last step we perform the Pattern Search. Also, one can try to improve the behavior of the Pattern Search by adding some directions using data from the model.

As in the Pattern Search method, if the iteration is successful we can increase the step size parameter; otherwise, we must decrease it.

Figure 3 describes the flowchart of the method.

3.4 Golden Section

The Golden Section Search method is an one-dimensional derivative-free optimization method [21]. This method shrinks an initial interval $[a, b]$, in which it is known to have a local minimizer of the objective function f.

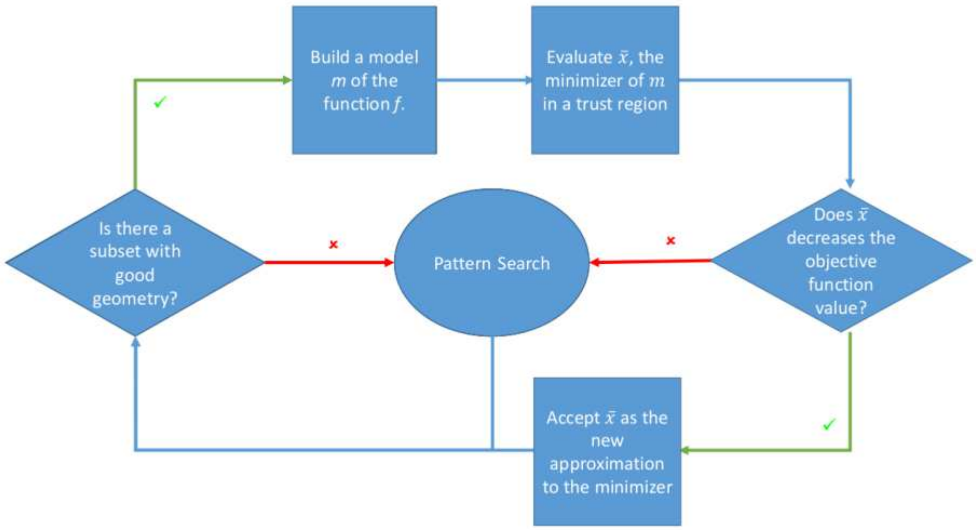


Fig. 3: Flowchart of the SID-PSM method.

At each iteration, we compute the points c and d (where $c < d$) that divides the original interval by the golden ratio, then if c gives a better objective function than d, we set $b = d$, and restart the process, otherwise we update a making it equals to c.

The reason to use the golden ratio is that the point not used in the update process will be the c or d of the next iteration, so we must evaluate the objective function in only one point at each iteration. The stop criterion for this method is usually used as the distance between a and b.

As mentioned before, this method is used to optimize one-dimensional problems. Here, our target refers to multidimensional problems. However, the method was used as an alternative strategy. In order to apply this method, we optimized each variable at a time. Also, it is important to notice that there is no mathematical guarantee that this strategy will converge to a stationary point.

4 Case study

4.1 Material & Methods

Experiments were carried out in a pilot farm situated near the city of Cabreuva, State of São Paulo, South-eastern Brazil. The broiler house has the following specifications: 150 m of length, 15 m of width and 2.5 m of height. $24,000$ birds (Cobb breed) in average are rearing there in a production cycle with an estimated duration of 42 days. The facility has ventilation system which contains 9 exhaust fans, an automatic controller and other devices which help on the environmental control inside the house.

Four temperature/relative humidity sensors are strategically placed in the facility, three of them to register the inside temperature and one of them to register the outside temperature, respectively. These sensors transmit the temperature values at each 5 minutes which are recorded in a simple datasheet of type csv or txt.

For this research a total of 10803 valid values were used to perform the experiments. The datasheet has 3 columns labeled as: Number of exhaust fans (number of fans), average of temperature in Celsius degrees from the three inside sensors (int temp $°C$) and outside temperature in Celsius degrees (ext temp $°C$). The information corresponding to each instant t (rows) involves the average number of exhaust fans turned on in the interval between t and $t + 5$ minutes, the average internal temperature considering 3 different positions of sensors in the house, and the external temperature at instant t. Table 1 presents a view of the data considering a sample of 6 rows.

Tab. 1: A Datasheet sample.

Number of fans	int. temp. $°C$	ext. temp. $°C$
0.57	29.57	23.12
0.23	29.26	22.12
0.13	29.49	21.80
0.13	29.74	21.50
0.12	29.89	21.50
0.14	30.11	21.86

For instance, in the first line, Table 1 presents the average internal temperature 29.57 $°C$, the external temperature 23.12 $°C$ and one of the exhaust fan was turned on 57% of the time (2 minutes, 51 seconds).

Due to that ideal thermal conditions (ideal temperature) depend on the infrastructure of the facility, age and bird breed, the original datasheet was divided into three subsets as follows: Matrix A (datasheet A) with 3601 rows which corresponds to the first third of the broilers' life, Matrix B with 3601 rows which corresponds to the second third of the broilers' life and, finally, Matrix C with 3601 rows which correspond to the last third of the broilers' life. Therefore, each of the matrices A, B and C involve 300 hours (or 12.5 days).

Three sets of parameters were estimated corresponding to the three periods of the bird's life.

Therefore, the four strategies of derivative-free optimization are implemented in the three datasets. The process began by giving trial values of the parameters p_1, p_2, p_3. Then, the first objective function was evaluated in the following way: We ran the PDE (Partial Differential Equation) model described in [11] from $t = 0$ to $t = 60$ minutes employing the real internal temperature at $t = 0$ as initial condition, and we call $T_{model}(60)$ as the average temperature at minute 60. Then, we ran the PDE model from $t = 60$ to $t = 120$ using the real internal temperature at $t = 60$ as initial condition, calling $T_{model}(120)$ as the average temperature computed by the model at minute 120. The observed average internal temperature provided by matrix A at minutes 60 and 120 are called $T_{obs}(60)$ and $T_{obs}(120)$, respectively. We proceed in the same way in the interval times $[120, 180]$, $[180, 240]$, $\ldots$, $[17940, 18000]$.

For these calculations we used the trial parameters, the data given by external temperatures, and the exhaust fan information given in A. The objective function value is the sum of squares of the differences between the predicted temperatures $T_{model}(t)$ and the observed temperature $T_{obs}(t)$, divided by the number of predicted temperatures, in this case 300. In a similar way we computed the objective functions that correspond to the second and third part of the data.

Even though the temperatures for every 5 minutes were available, only the temperatures at each hour were used at the objective function. This choice was made considering that the PDE model is proposed as a tool to predict the internal temperatures for a reasonably large time interval. If the parameters are optimized using the predictions for every 5 minutes, they can be not properly fitted for predictions of larger time intervals.

Let us emphasize that the PDE model is restarted for every 60 minutes of simulation, that is, to predict the internal temperature at instant 60, we use the real internal temperature at instant 0, to predict the temperature at instant 120 we use the real one at instant 60, and so on. Finally, we estimated another set of parameters, corresponding to the whole broilers' life. The corresponding data set was stored at matrix D.

Pattern Search was executed by using the routine "patternsearch" (default parameters, except for tolerance) in MatLab. "SID-PSM version 2.0" (default parameters, except for tolerance and the step of the incremental parameter, in this case value of 2) in MatLab for the SID-PSM method. BOBYQA was executed in FORTRAN (default parameters, except for the tolerance) and Golden Section was implemented in MatLab by setting the distance between consecutive points to define the stopping criterion. For the whole set of strategies, the parameter that defines the stopping criterion was set to 10^{-8}.

4.2 Results and discussion

Results of the optimization for each training set and each method are presented in Tables 2 to 5. The improvement with respect to the initial approximation given by $p_1 = 1, p_2 = 0.5, p_3 = 1$, is also shown. The initial approximation has been obtained from [11] by trial and error.

Tab. 2: Optimization of $m(p)$ using A. This table shows the function optimal value, the optimal parameters obtained, the improvement with respect to the initial approximation (%) and the number of function evaluations performed by each method.

Method	$m(p)$	p_1	p_2	p_3	%	`fevals`
Initial approx.	1.50	1.00	0.50	1.00	0.0	–
Pattern Search	1.29	4.39	40.00	1.19	14.1	491
SID-PSM	1.29	4.39	40.00	1.19	14.1	513
BOBYQA	1.29	4.39	40.00	1.19	14.1	103
Golden Section	1.33	2.48	40.00	2.04	11.7	144

From Tables 2 to 5 we derive the following conclusions.

1. All the optimization methods were successful in terms of improving the approximation given by trial-and-error.

Tab. 3: Optimization of $m(p)$ using B. This table shows the function optimal value, the optimal parameters obtained, the improvement with respect to the initial approximation (%) and the number of function evaluations performed by each method.

Method	$m(p)$	p_1	p_2	p_3	%	`fevals`
Initial approx.	2.13	1.00	0.50	1.00	0.0	–
Pattern Search	1.06	18.76	40.00	0.34	50.2	1140
SID-PSM	1.06	18.76	40.00	0.34	50.2	716
BOBYQA	1.06	18.76	40.00	0.34	50.2	184
Golden Section	1.09	16.67	40.00	0.47	48.7	144

Tab. 4: Optimization of $m(p)$ using C. This table shows the function optimal value, the optimal parameters obtained, the improvement with respect to the initial approximation (%) and the number of function evaluations performed by each method.

Method	$m(p)$	p_1	p_2	p_3	%	`fevals`
Initial approx.	4.00	1.00	0.50	1.00	0.0	–
Pattern Search	1.99	13.07	40.00	0.31	50.2	1088
SID-PSM	1.99	13.07	40.00	0.31	50.2	577
BOBYQA	1.99	13.07	40.00	0.31	50.2	107
Golden Section	2.02	11.20	40.00	0.39	59.7	144

2. All the methods found the same solutions, except Golden Section, that obtained poorer approximations.

3. BOBYQA was the most efficient method since it always obtained the lowest functional values and the smallest number of functional evaluations.

4. The optimal parameter p_2 always reached its upper bound. This means that, in the best case, the model runs essentially "without walls". This is a limitation of the one-dimensional formulation.

Using the fourth training set (D) as an example, the function values

Tab. 5: Optimization of $m(p)$ using D. This table shows the function optimal value, the optimal parameters obtained, the improvement with respect to the initial approximation (%) and the number of function evaluations performed by each method.

Method	$m(p)$	p_1	p_2	p_3	%	`fevals`
Initial approx.	2.54	1.00	0.50	1.00	0.0	–
Pattern Search	1.50	9.43	40.00	0.35	41.2	903
SID-PSM	1.50	9.43	40.00	0.35	41.2	506
BOBYQA	1.50	9.43	40.00	0.35	41.2	153
Golden Section	1.51	8.76	40.00	0.43	40.7	144

versus the number of iterations for each method was plotted (Figure 4).

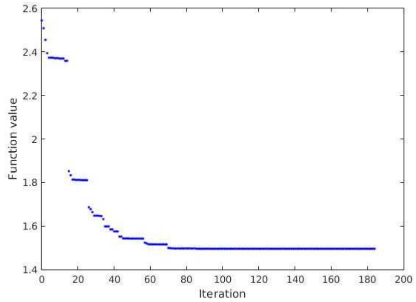
In Figure 4, it is possible to observe that the optimal point is obtained at the beginning of the iterative processes; most iterations are used to confirm that the point is optimal.

In order to improve the understanding of the results, some profiles of the objective functions were plotted. For this, we fixed two variables as the optimum values found by SID-PSM and let one to vary. In Figure 5, the profiles using A are presented while in Figure 6, profiles using Matrix B are shown.

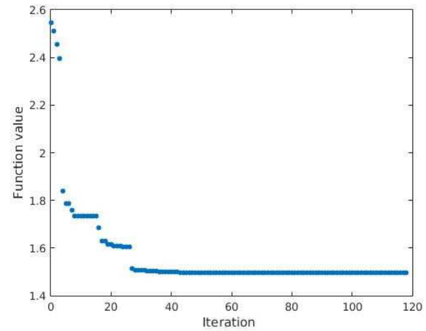
These profiles show that m, as a function of p_2, tends asymptotically to a horizontal line and the unconstrained optimization problem has no (global) minimizer. Also, it can be seen that m is well-behaved and seems to have no stationary points other than the solution of the problem. Although these graphics were plotted using the fixed variables as the optimal values found using SID-PSM, similar behavior is obtained when they are fixed as the optimal values found by Pattern Search or BOBYQA.

5 Conclusions

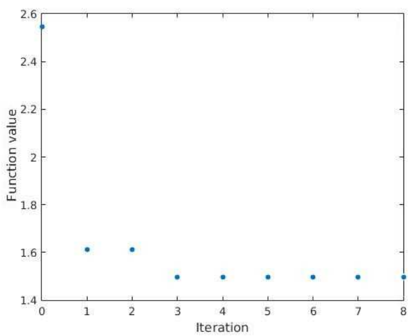
The real-time features of the temperature control model make it necessary to select the fastest method for parameter estimation and fitting. Correction of the parameters (p_1, p_2 and p_3) should occur during the operation of the system and consequently, speed (computing time) is crucial for compatibility



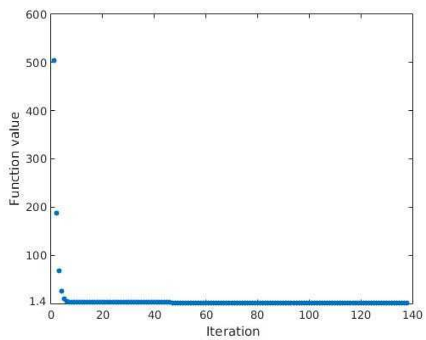
(a) Pattern Search.



(b) SID-PSM.



(c) BOBYQA.



(d) Golden Section.

Fig. 4: Graphics of the function value per iteration for different methods using the fourth training set.

with the growing, rearing and dynamics process of the broilers by itself. As was mentioned before, the control of thermal environment inside the broiler houses is a complex problem. A strategy related to impose fixed parameters for consecutive execution in real-time of the model is unfeasible and unreliable. So, self-optimization and incorporation of the estimation/fitting software into the control model is crucial for the effective and accurate response of the system.

In this sense and after being evaluated the results of the four strategies with derivative-free methods, Powell's software BOBYQA seems to be the most adequate tool for this type of parameter estimation. The reason for the superiority of BOBYQA is that, as shown in figures 5 and 6, the objective function is well-behaved and does not seem to present attractive stationary points, other than the solutions of the problem. BOBYQA proceeds interpolating quadratic models, so its comparative performance improves when the function is smooth and presumably unimodal. The derivatives of the objective function with respect to the parameters are computable, either by hand-calculation or by automatic differentiation. However, the human cost of these tasks are rather discouraging, especially at a level of development in which one changes frequently the structure of the model. The friendliness of derivative-free software makes it preferable to more sophisticated alternatives. Moreover, the commercial feasibility of temperature control model imposes that our own solver for real time optimization must be developed. Thus, the present research allowed detecting the best type of software and adjustments that should be implemented.

In addition to the decision on the best solver for parameter estimation, the present numerical study provided the understanding of the structure of the problem. The fact that the best wall diffusion parameter is infinity revealed that something is inadequate in the formulation of the PDE model. This is not surprising because the PDE model comes from a radical one-dimensional simplification of a Fluid Mechanics problem whose solution in real time is impossible, at least subject to the budget restrictions of this project in terms of computing time. Since 3D models are certainly unaffordable, it can be conjectured that 2D models should be developed. Our present feeling is that models based on a parallel plane to the floor could reflect adequately the diffusion through different types of walls.

Acknowledgements

This work was supported by PRONEX-CNPq/FAPERJ E-26/111.449/2010-APQ1, FAPESP (grants 2010/10133-0, Cepid-Cemeai 2011-51305-02, 2013/03447-6, 2013/05475-7, and 2013/07375-0), and CNPq.

References

[1] A. Donkoh. Ambient temperature: a factor affecting performance and physiological response of broiler chickens. *International Journal of Biometeorology*, 33(4):259–265, 1989. doi: 10.1007/BF01051087.

[2] R. Barnwell and A. Rossi. Maximização da performance

em períodos quentes. *Avicultura Industrial*, 11:72–80, 2003. http://www.aviculturaindustrial.com.br.

[3] A.G. Amaral. Effect of the production environment on sexed broilers reared in a commercial house. *Arq. Bras. Med. Vet. Zootec.*, 63(3):649–658, 2011. doi: 10.1590/S0102-09352011000300017.

[4] D. Renaudeau, A. Collin, S. Yahav, V. Basilio, J.L. Gourdine, and R.J. Collier. Adaptation to hot climate and strategies to alleviate heat stress in livestock production. *Animal*, 6(5):707–728, 2012. doi: 10.1017/S1751731111002448.

[5] W.M. Razuki, S.A. Mukhlis, F.H. Jasim, and R.F. Hamad. Productive performance of four commercial broilers genotypes reared under high ambient temperatures. *International Journal of Poultry Science*, 10(2):87–92, 2011. doi: 10.3923/ijps.2011.87.92.

[6] E. Bustamante, F.J. García-Diego, S. Calvet, F. Estellés, P. Beltrán, A. Hospitaler, and A.G. Torres. Exploring ventilation efficiency in poultry buildings: The validation of computational fluid dynamics (CFD) in a cross-mechanically ventilated broiler farm. *Energies*, 6(5):2605–2623, 2013. doi: 10.3390/en6052605.

[7] M. Reboiro-Jato, J. Glez-Dopazo, D. Glez, R. Laza, J. F. Galvez, R. Pavon, D. Glez-Pena, and F. Fernandez-Rivarola. Using inductive learning to assess compound feed production in cooperative poultry farms. *Expert Systems with Applications*, 11:14169–14177, 2011. doi: 10.1016/j.eswa.2011.04.228.

[8] F. Rojano, P.E. Bournet, M. Hassouna, P. Robin, M. Kacira, and C.Y. Choi. Modelling heat and mass transfer of a broiler house using computational fluid dynamics. *Biosystems Engineering*, 136(5):25–38, 2015. doi: 10.1016/j.biosystemseng.2015.05.004.

[9] M.S. Baracho, I.A. Nääs, G.R. Nascimento, J.A. Cassiano, and K.R. Oliveira. Surface temperature distribution in broiler houses. *Biosystems Engineering*, 13(3):177–182, 2011. doi: 10.1590/S1516-635X2011000300003.

[10] I.A. Nääs. Uso de técnicas de precisão na produção animal. *Revista Brasileira de Zootecnia*, pages 358–364, 2011.

http://livrozilla.com/doc/816069/revista-brasileira-de-zootecnia-uso-de-t%C3%A9cnicas-de-precis...

[11] D.T. Detsch. *Modelo com aprendizagem automática para previsão e controle de temperatura em aviários tipo túnel de vento*. PhD thesis, Department of Applied Mathematics, Institute of Mathematics, Statistics and Scientific Computing, University of Campinas, 2016. http://repositorio.unicamp.br/jspui/handle/REPOSIP/307474.

[12] C. Audet and D. Orban. Finding optimal algorithmic parameters using derivative-free optimization. *SIAM Journal on Optimization*, 17(3):642–664, 2006. doi: 10.1137/040620886.

[13] B.H. Cervelin. Sobre um método de otimização irrestrita baseado em derivadas simplex. Master's thesis, Department of Applied Mathematics, Institute of Mathematics, Statistics and Scientific Computing, University of Campinas, 2013. http://repositorio.unicamp.br/handle/REPOSIP/306038.

[14] S.M. Wild, J. Sarich and N. Schunk. Derivative-free optimization for parameter estimation in computational nuclear physics. *Journal of Physics G: Nuclear and Particle Physics*, 42(3), article id. 034031, 2015. doi: 10.1088/0954-3899/42/3/034031.

[15] P.K. Murkherjee, I.; Ray. A review of optimization techniques in metal cutting process. *Computers and Industrial Engineering*, 50(1-2):15–34, 2006. doi: 10.1016/j.cie.2005.10.001.

[16] M.J.D. Powell. Beyond symmetric broyden for updating quadratic models in minimization without derivatives. *Mathematical Programming*, 138(1-2):475–500, 2013. doi: 10.1007/s10107-011-0510-y.

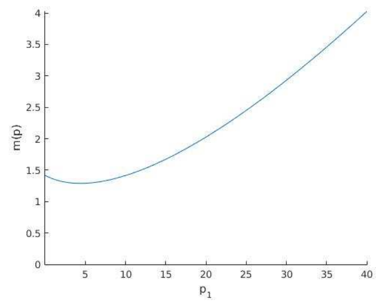
[17] M.J.D. Powell. On the convergence of trust region algorithms for unconstrained minimization without derivatives. *Computational Optimization and Applications*, 53(2):527–555, 2012. doi: 10.1007/s10589-012-9483-x.

[18] M.J.D. Powell. The bobyqa algorithm for bound constrained optimization without derivatives. Technical Report Cambridge NA Report NA2009/06, University of Cambridge, Cambridge, august 2009. http://www.damtp.cam.ac.uk/user/na/NA_papers/NA2009_06.pdf.

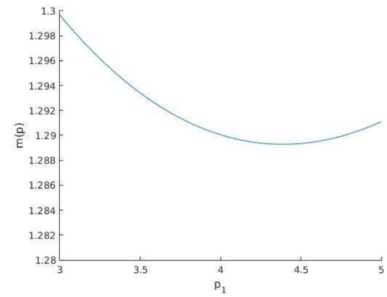
[19] R. M. Lewis and V. Torczon. Pattern search algorithms for bound constrained minimization. *SIAM Journal on Optimization*, 9(4):1082–1099, 1999. doi: 10.1137/S1052623496300507.

[20] A.L. Custódio and L.N. Vicente. Using sampling and simplex derivatives in pattern search methods. *SIAM Journal on Optimization*, 18(2):537–555, 2007. doi: 10.1137/050646706.

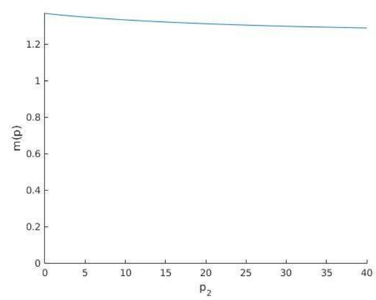
[21] Y. Luenberguer, D.G.; Ye. *Linear and Nonlinear Programming*. International Series in Operations Research & Management Science. Springer, 2009. ISBN 978-3-319-18842-3.



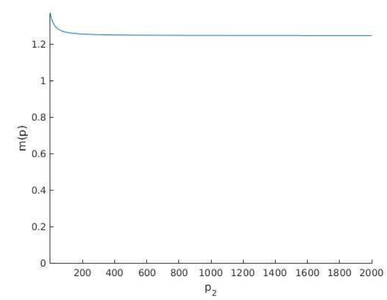
(a) The average squared error as function of p_1 has a unique minimizer at $p_1 = 4.39$.



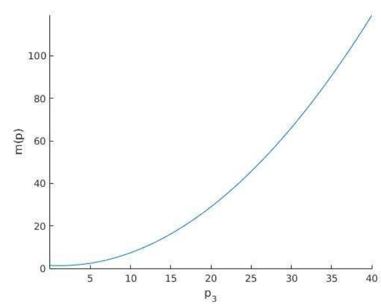
(b) The average squared error as function of p_1 in a smaller interval, p_1 varying from 3 to 5.



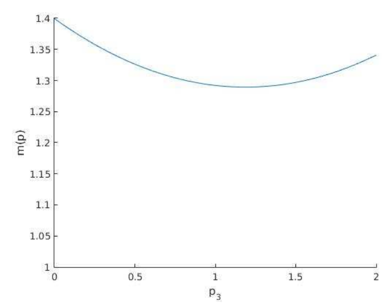
(c) The average squared error as function of p_2 has no minimizer, it tends asymptotically to a horizontal line.



(d) The average squared error as function of p_2 in a larger interval showing how the error tends to an horizontal line.

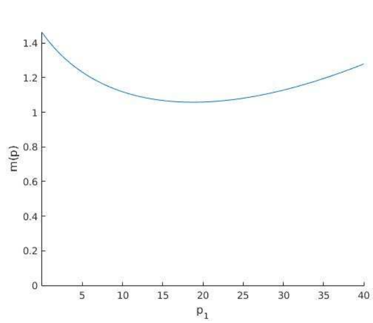


(e) The average squared error as function of p_3 has a unique minimizer at $p_3 = 1.19$.

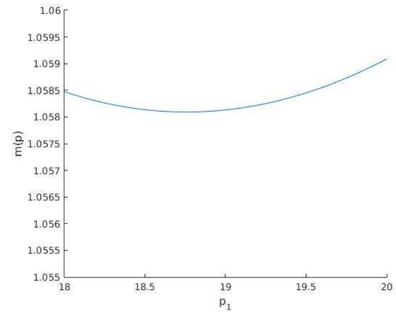


(f) The average squared error as function of p_3 in a smaller interval, p_3 varying from 0 to 2.

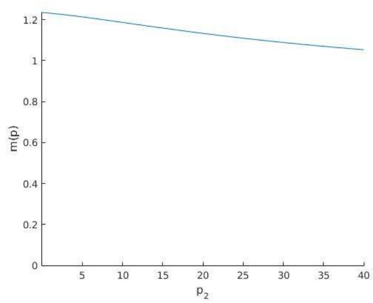
Fig. 5: Profiles using Matrix A, these graphics have two variables set as the optimum value found by SID-PSM and the other one is free. They show how the average error behaves when we change just one of the variables.



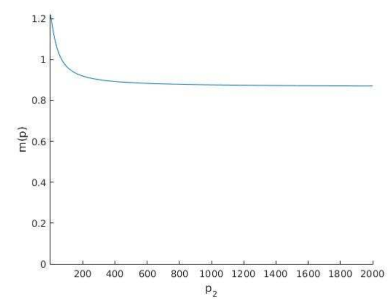
(a) The average squared error as function of p_1 has a unique minimizer at $p_1 = 18.76$.



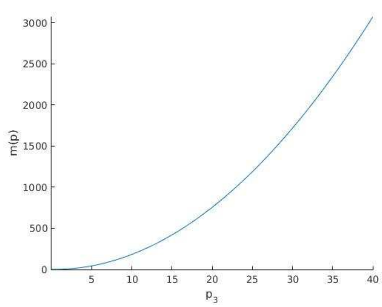
(b) The average squared error as function of p_1 in a smaller interval, p_1 varying from 18 to 20.



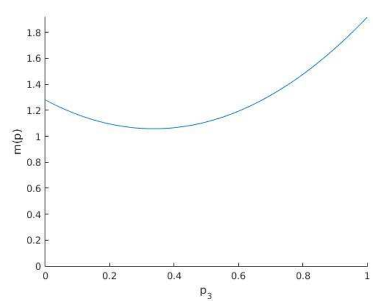
(c) The average squared error as function of p_2 has no minimizer, it tends asymptotically to a horizontal line.



(d) The average squared error as function of p_2 in a larger interval showing how the error tends to an horizontal line.



(e) The average squared error as function of p_3 has a unique minimizer at $p_3 = 0.34$.



(f) The average squared error as function of p_3 in a smaller interval, p_3 varying from 0 to 1.

Fig. 6: Profiles using Matrix B, these graphics have two variables set as the optimum value found by SID-PSM and the other one is free. They show how the average error behaves when we change just one of the variables.