Comparing modeling alternatives for solving a non-linear single Hydro Unit Commitment problem

Alexandre Heintzmann, Christian Artigues, Pascale Bendotti, Sandra Ulrich Ngueveu, Cécile Rottner

Abstract

A wide range of real world optimization problems involves continuous decisions and non-linearities. Each non-linear component of such problems can be modeled either linearly or non-linearly, considering or not additional integer variables. This results into different modeling choices that can drastically impact the solution time and quality. In this paper, we evaluate representative modeling alternatives, including common models from the literature as well as new models featuring less common functions. The single Hydro Unit Commitment problem (1-HUC) is a non-linear use case considered. The non-linearities involved come from the power produced. The power is defined as a two-dimensional non-convex and non-concave function of the water flow and head decision variables, the latter being itself a one-dimensional concave function of the turbined volume. We consider both the general problem and a common special case, assuming that the water head is fixed. Several available solvers are used for each non-linear model and the best virtual solver is retained to focus on the model capabilities rather than on the solver performance. Based on the numerical experiments, three models stand out as the most efficient in terms of computational time, solution quality and feasibility, sometimes in a counter-intuitive manner. For each of these models, a solver is highlighted as the most adequate.

Comparing modeling alternatives for solving a non-linear single Hydro Unit Commitment problem

Alexandre Heintzmann[1,2], Christian Artigues[2], Pascale Bendotti[1], Sandra Ulrich Ngueveu[2], and Cécile Rottner[1]

[1]EDF Lab Paris-Saclay, 7 Bd. Gaspard Monge, 91120 Palaiseau, France
{alexandre.heintzmann, pascale.bendotti, cecile.rottner@edf.fr}
[2]LAAS-CNRS, Université de Toulouse, CNRS, INP, Toulouse, France
{alexandre.heintzmann, christian.artigues, sandra.ulrich.ngueveu@laas.fr}

February 17, 2023

1 Introduction

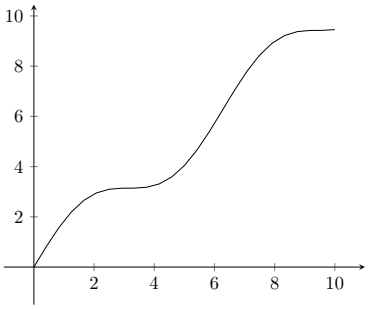
In the real world, systems involving continuous decisions and non-linearities are frequent. In the literature, optimizing such systems via mathematical programming requires to choose between two main modeling alternatives of each non-linear component, namely either a linear or a non-linear model, yielding possibly additional integer variables. A non-linear model usually represents more closely a physical system than a linear one, but tends to require a longer computing time, in particular when no convexity property applies. Within these two main modeling possibilities, there are still a lot of modeling choices to make. For illustration purposes, **Figure 1** shows a real-world continuous non-linear function on interval $[0; 10]$ and four alternative functions, amongst many others, to model it. Figure **(1a)** shows the real-world continuous non-linear function. Alternative **(1b)** relies on a single non-linear function, alternative **(1c)** uses a family of elementary non-linear functions, alternative **(1d)** is based on a piecewise-linear (PWL) function, and alternative **(1e)** considers a finite set of discrete points. These alternatives have several differences, such as the type of function(s) involved: non-linear non-convex non-concave for **(1b)**, concave for **(1c)** and linear for **(1d)**, or the need of additional binary variables for **(1c)**, **(1d)** and **(1e)**.

Thus, two alternatives can have different properties, meaning their resolution may be nothing alike. Faced to a real-world non-linear problem, choosing the best modeling alternative is a crucial issue.

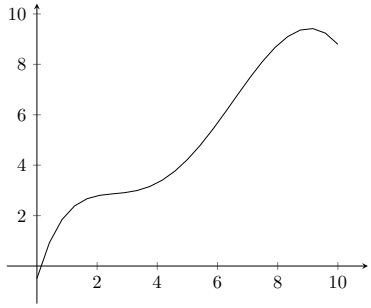
The non-linear use case considered in this paper is the single unit Hydro Unit Commitment (1-HUC). The 1-HUC features two non-linearities, namely a one dimensional concave non-linearity, and a two dimensional non-convex non-concave non-linearity. Such non-linearities can lead to various issues, which can be highlighted using different indicators, such as precision, feasibility, computing time or solution quality. The modeling of the hydroelectric power function already raised interest in the literature, such as in [12] where three different models are compared for the hydropower maintenance scheduling, with a focus on the approximation of the hydroelectric power function.

In this paper, we push the study of the impact of non-linear modeling further. First of all, the power functions considered in the literature are usually simplifications, which may not reflect all the non-linear features of the power function. As such, we define a new power function closer to the physics, based on the characteristics of the power function for each turbine, assuming a fixed turbine start-up sequence. As the model featuring this power function is out of reach for current solvers, seven modeling alternatives are described for the 1-HUC. These seven models cover a large panel, ranging from linear to non-linear models, with and without integer variables, and are representative of how the literature handles the approximation functions described in **Figure 1**. More precisely, four models are proposed for with a unique function represented by **Figure 1b**, one for each of the alternatives. As any two non-linear solvers do not implement the same tools, they are expected to behave differently. This is why the proposed non-linear models will be solved using five black-box global optimization solvers available on Neos Server [9]. The principle is to evaluate these models for the 1-HUC using the afore-mentioned indicators and to identify the characteristics of the 1-HUC instances impacting their solution time and quality. The main contribution of the paper is to make general modeling recommendations from the numerical experiments, depending on the characteristics of the instance, the desired precision and the allowable computing time.

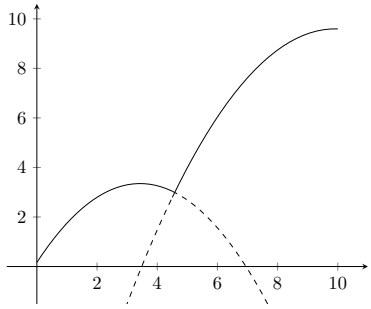
In Section 2 a literature review of solution approaches for non-linear optimization problems is proposed. In Section 3, the 1-HUC is defined, followed by a literature review on the non-linearities of the HUC. In Section 4 the proposed models are described and compared from a theoretical point of view. In Section 5, numerical experiments illustrate the comparative performance of the models on different sets of realistic 1-HUC instances. In Section 6, concluding remarks and perspectives for further research are drawn.



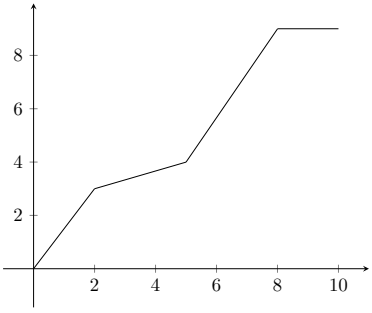
(a) Real-world non-linear function



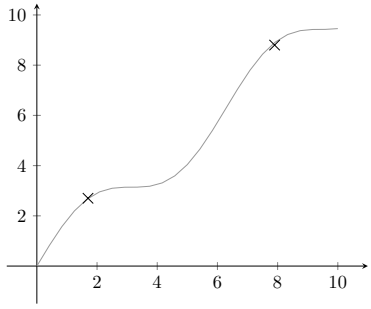
(b) Approximation with a non-linear function



(c) Approximation with a family of elementary functions



(d) Approximation with a PWL function



(e) Approximation with a discrete set of decisions

Figure 1: Four different approximations of a real-world non-linear function

2 Solution approaches for non-linear optimization problems

As aforementioned, to optimize a non-linear system using mathematical programming, the main possibilities are to solve either a linear or a non-linear model. Linear models lead to Mixed Integer Linear Programs (MILP), and non-linear models to Non-Linear Programs (NLP) or Mixed Integer Non-Linear

Programs (MINLP). For these three types of programs, exact algorithms are based on a divide and conquer strategy. In all cases the search space is divided in order to obtain lower and upper bounds in each sub-space.

For a minimization problem, the upper bound is derived from any feasible solution, and the lower bound is obtained by solving a relaxation of the problem. For an MILP, the relaxed problem consists in ignoring the integrity constraints [19]. For an NLP and an MINLP, the relaxed problem is obtained by ignoring the integrity constraints and via convex under-estimators [34]. In this section, the main modeling techniques are described, along with state-of-the-art solution methods and more specifically the ones implemented in the non-linear solvers used for the numerical experiments. Note that, although the literature on the 1-HUC mostly focuses on heuristics, we restrict our study to the exact methods implemented in the selected solvers, thus allowing us to enlight the full potential of each tested model when optimality can be proven.

2.1 Modeling

Modeling a non-linear problem as an MILP is usually driven by the will to have smaller computing times. The functions approximating non-linearities within an MILP are PWL as shown in **Figure (1d)**. A PWL function is defined as a collection of affine functions on intervals. The bounds of the intervals, as shown in **Figure (1d)** are called breakpoints. From the literature, several PWL formulations exist. The authors in [8] and [15] compare three classical formulations, convex combination, multiple choice and incremental, which are proven to be equivalent in terms of relaxation. The standard approach for these formulations is to approximate a non-linear univariate function, but there are extensions for two-dimensional functions. For example in [10] three methods to approximate two-dimensional non-linear functions where breakpoints exist in both dimensions are presented. There are also results to approximate a function for any dimension, in [15], where a generalized PWL model for any dimension is proposed. These PWL formulations can be improved in order to reduce the number of variables required. For instance introducing disjunctive constraints makes it possible to describe a PWL model with a logarithmic number of variables as proposed in [37]. Another example is to use a non-necessarily continuous PWL function. In the case of an univariate function, it is proven in [25] that for a given precision, a PWL function can be derived with less pieces if the PWL function is non necessarily continuous at the breakpoints.

Modeling a non-linear problem as an (MI)NLP is usually driven by the will to have a better representation of the physics. If the (MI)NLP is convex, the problem admits a single local optimum, which is also global. Also, the relaxation only consists in relaxing the integrity constraints, as the problem is convex, it is itself a convex under-estimator. However, if the (MI)NLP is non-convex, global optimization algorithms are required. The relaxation then relies on both relaxing the integrity constraints, and finding a convex under-estimator. If the functions of a non-convex (MI)NLP have an efficient under-estimator, a better lower-bound will be produced, meaning faster convergence. For this purpose,

the survey in [5] describes under-estimators for various non-linear functions. To help finding better convex under-estimators, techniques such as the symbolic reformulation in [34] and the bound tightening [3] can be performed. In all cases, the modeling choices can alter the feasibility of the problem. With high exponents or functions like $log(x)$ with x close to 0, approximation errors can be induced, in particular in floating point computing.

Pointing out the differences between the family of the models is also important because the tools used to solve a problem depends on its structure. To solve an MILP, the well known Branch and Bound (BB) algorithm [19] and its derivatives the Branch and Cut [27], the Branch and Price [32] and the Branch and Cut and Price [11] can be used. To solve a convex (MI)NLP, as the relaxation is similar to a MILP, a variant of the BB algorithm can be used. To solve a non-convex (MI)NLP, a global optimisation algorithm is required. The main algorithm involved in most global optimization tools is the spatial Branch and Bound (sBB) [34], designed to solve an MINLP. If modeling requires no integer variables, the model remains continuous (**Figure (1b)**) and results in an NLP. Variations of the sBB for NLP can be used, such as the α Branch and Bound, the Reduced Space Branch and Bound [13], or the Branch and Contract [39]. If modeling requires additional integer variables (**Figure (1c)**), the algorithms involved in the tool must be able to handle an MINLP. As aforementioned, the sBB can solve this type of models as well as other algorithms such as the Branch and Reduce [30], the sBB with interval analysis [36] or the GMIN and SMIN algorithms [1].

2.2 Solvers

Five non-linear solvers are considered in this paper, ANTIGONE [24], BARON [35], COUENNE [2], LINDOGlobal [22] and SCIP [38], as they are accessible on Neos Server [9] with GAMS format. All of the five solvers implement global optimization algorithms, and use the sBB algorithm or its derivatives. To improve the solution time and quality, the solvers complement variations of the sBB with different tools. However, the set of implemented tools is not the same from a solver to another, which means that their performance may vary depending on the model.

Solver CPLEX [7] is used to solve MILPs in this article. As comparing linear solvers is beyond the scope of this work, a well known and efficient linear solver is arbitrarily chosen. Clearly non-linear solvers can also solve linear models. However linear solvers are dedicated to linear problems, and give a more appropriate reference, in particular for a comparison with the resolution of non-linear models.

The six solvers are further described in **Appendix A**.

3 1-Hydro Unit Commitment

3.1 Definition of the problem

The 1-HUC is defined as follows. Consider a valley containing one unit located between a single upstream reservoir and a single downstream reservoir. From the hydroelectric production principle, the water from the upstream reservoir flows through the unit to the downstream reservoir, operating the turbines of the unit, thus generating electric power. The time horizon is discretized in T time periods, each of duration Δ. At each time period t the water flow d_t going through the unit must lie within the interval $[\underline{D}, \overline{D}]$. The power p_t produced at the time period t depends on the water flow d_t, but also on the reservoir head h_t. The head h_t is the height difference between the surface of the water in the upstream reservoir and the unit, and depends on the volume of the upstream reservoir v_t^1. Each reservoir $n \in \{1, 2\}$ has a maximum capacity $\overline{V}_t^n$ and minimum capacity $\underline{V}_t^n$, variable through time. Variable maximum and minimum capacity makes it possible to set target volumes, when $\overline{V}_t^n = \underline{V}_t^n$. At each time period, the reservoir n has an additional intake of water A_t^n, which can be positive (rain, melting snow) or negative (use of water for surrounding agriculture). At each time period, we consider the energy to be sold at forecasted unitary price Λ_t, variable through the time. At the end of time period T, the water in reservoir n has an expected unitary value Φ^n. Value Φ^n is the expected value of the energy produced using the water. A higher Φ^n will lead to preserve more water and produce less energy, and the other way around for a lower Φ^n. The HUC considered is a revenue-maximizing price-taker scheduler problem, where the power prices, the water expected value and the reservoir external inflows and reservoirs capacities are parameters. There exist other types of HUC, for instance where the aim is for the energy produced to meet the demand [14].

The profit takes into account the total value of the water in each reservoir at the end of time period T, and the value of the energy produced. Solving the HUC consists in maximizing the profit, while satisfying the capacities and the target volumes at each time period. A generic model (P_{gen}) can be defined, using the water flow d_t, the power p_t, the volume in the upstream reservoir v_t^1 and in the downstream reservoir v_t^2 and the head of the upstream reservoir h_t as decision variables. Function $\boldsymbol{f}$ is the one-dimensional concave function, giving the water head h_t at time period t with respect to the volume and function $\boldsymbol{F}$ is the two-dimensional non-convex and non-concave function, representing the power of the unit, depending on the water flow d_t and the head h_t. Both functions $\boldsymbol{f}$ and $\boldsymbol{F}$ are considered as nondecreasing and nonnegative. Model

(P_{gen}) is given as follows:

$$\max \sum_{t=1}^{T} \Delta \cdot \Lambda_t \cdot p_t + \sum_{n=1}^{2} \Phi^n \cdot v_T^n \tag{1}$$

$$\text{s.t. } v_t^1 = V_0^1 + \sum_{t'=1}^{t} \left(A_{t'}^1 - d_{t'} \Delta \right) \qquad \forall t \leq T \tag{2}$$

$$v_t^2 = V_0^2 + \sum_{t'=1}^{t} \left(A_{t'}^2 + d_{t'} \Delta \right) \qquad \forall t \leq T \tag{3}$$

$$h_t = \boldsymbol{f}(v_t^1) \qquad \forall t \leq T \tag{4}$$

$$p_t = \boldsymbol{F}(d_t, h_t) \qquad \forall t \leq T \tag{5}$$

$$\underline{V}_t^n \leq v_t^n \leq \overline{V}_t^n \qquad \forall t \leq T, \forall n \in \{1, 2\} \tag{6}$$

$$\underline{D} \leq d_t \leq \overline{D} \qquad \forall t \leq T \tag{7}$$

Constraints (2) and (3) are volume conservation constraints. Constraints (4) computes the water head h_t, using the concave function $\boldsymbol{f}$ of the volume. Constraints (5) computes the power p_t, using the non-convex non-concave non-linear function $\boldsymbol{F}$ of the water flow d_t and the head h_t. Constraints (6) and (7) give upper and lower bounds for variables. The criterion to maximize is the profit, which is a linear expression described by (1).

A standard simplification of the 1-HUC is to assume a constant head $h_t = H$, which leads to the fixed-head 1-HUC. This simplification is relevant for some instances of the 1-HUC where volume variations are small enough for the impact on the unit efficiency to be insignificant. As such, inequalities (4) and (5) from (P_{gen}) are replaced by:

$$p_t = \boldsymbol{F}(d_t, H) \qquad \forall t \leq T \tag{8}$$

For the fixed-head 1-HUC, function $\boldsymbol{F}$ becomes a one-dimensional function, but remains non-convex non-concave. Note that even if the head is constant, we still consider variable v_t^1 and v_t^2 in the model, to ensure that reservoir capacities are respected.

3.2 Literature of the non-linearities of the HUC

As mentioned in [12], there are cases of the HUC where no perfect analytic representation of the hydroelectric power function is known. Nevertheless, some functions have been described as a baseline to measure the approximation error of the various models proposed in the literature. A generic hydroelectric power function [16] is:

$$p_t = \rho \cdot G \cdot h_t \cdot d_t \tag{9}$$

The power p_t is the product of ρ the density of water, G the gravitational constant, d_t the water flow and h_t the head. With $h_t \geq 0$ and $d_t \geq 0$, this

generic bilinear function is concave. However, in various papers of the literature [20] [23] [4] [12] [26], the shape of function $\boldsymbol{F}$ is described as non-convex and non-concave. In [26] a more sophisticated function is provided:

$$p_t = C_1 \cdot (v_t^1)^2 + C_2 \cdot (d_t)^2 + C_3 \cdot v_t^1 \cdot d_t + C_4 \cdot v_t^1 + C_5 \cdot d_t + C_6$$

- with C_1 to C_6 being constants. Note that this function may lead to approximation errors depending on the characteristics of the turbines and their number, as they are not explicitly considered in the function. Another approach uses a grid to have a reference for the hydroelectric power function obtained by discretizing the water flow and the head [12]. An algorithm, described in [33], is used to obtain a set of such grids, each of them representing the power function for a given number of active turbines. For each point on one of these grids, the value of the hydroelectric power function is computed with a dynamic programming algorithm based on a bilinear function similar to (9). The resulting grids are overvaluations of the power function, meaning that they no not necessarily reflect its actual shape. In order to have a function as close to the physics as possible, we will propose in this paper a new analytic function, considering the power function of each turbine explicitly for a fixed turbine start-up sequence.

In terms of modeling, some (MI)NLP featuring (9) are described in the literature. Indeed, a bilinear function is a common non-linear function can be well handled by non-linear solvers, even for large-scale instances. In [23] the HUC considered has multiple units, and is modeled as an NLP. Compared to (P_{gen}), function $\boldsymbol{f}$ not only depends on the volume of the upstream reservoir but also on the volume of the downstream reservoir and function $\boldsymbol{F}$ is a bilinear function depending on the water flow and the head. In [21], an algorithm called spatial Hydro Branch and Bound (sHBB) has been developed to solve to optimality the HUC with cascading units. This algorithm is used to solve an MINLP, where the power is a bilinear function of both water head and water flow. Comparing again with (P_{gen}), function $\boldsymbol{f}$ is a polynomial function with degree 4, and function $\boldsymbol{F}$ is a bilinear function depending on the water flow and the head. All these non-linear formulations define the power as merely linear with respect to the water flow, meaning that function $\boldsymbol{F}$ is always a bilinear function.

Another common modeling alternative of the literature is to approximate the power function with a piecewise linear function. In [28] the 1-HUC is considered. The authors introduce a family of univariate PWL functions to model the power depending on the water flow. Each PWL function of this family represents, for a specific volume, the power with respect to the water flow using exactly 4 pieces. To compare to (P_{gen}), function $\boldsymbol{F}$ becomes one of the univariate PWL functions introduced. The model also approximates with PWL functions the spilled flow constraints (see definition below), which involve a polynomial function. The proposed model in [4] for the 1-HUC also expresses the power with a family of univariate PWL functions of the water flow, for specific volumes. Besides, it takes into account the maximum variation of the water flow between two consecutive time periods. An improvement of this model features the rectangle

method [10]. The aim is to compute a better approximation when v_t^1 is between two of the specific volumes selected to compute the PWL functions. To do so, the method computes a projection of the power between the two surrounding specific volumes in order to rectify the approximation. There are also iterative methods using PWL functions. In [14] the HUC with cascading units is considered. It is pointed out that if the head is fixed, then the power depends only on the water flow. Using a PWL function with two pieces, the procedure is to solve HUC with fixed head iteratively, while updating the head between each iteration until convergence. Besides PWL formulations, hyperplanes formulations have also been developed [29]. The aim is to create a set of hyperplanes, for each number of active turbines, to linearize the non-linear power function. More precisely, the hyperplanes are deduced from the most efficient point, and each set form a concave over-estimator of the power function for a given number of active turbines. As a maximization problem is considered, these hyperplanes yield a convex optimization region. Defining multiple sets aims to produce a more precise approximation, based on the aforementioned grid approximation of the power function for each number of active turbines [33]. For a given number of active turbines, the linearization does not require any additional binary variable. However, binary variables are required to indicate the number of active turbines at each time period, thus resulting in a MILP.

In [12] three models for the hydro power maintenance scheduling are compared. The three models involve respectively a formulation with hyperplanes [29], a PWL formulation from [10], and a five degree polynomial function. As mentioned previously, the grid approach [33] is considered as a baseline to compare the economic value of the solutions. The result of this comparison is that the hyperplanes formulation is overall the best compromise between the size and complexity of the optimization problem and the deviation from the reference data. As the grid approach is a simplification of the power function, there was no need to resort to more elaborated functions. In our study, we would like to make a similar comparison, but with a power function taking into account the power of each turbine explicitly. It becomes relevant to include more sophisticated models in this comparison, as they may lead to smaller approximation errors.

As we can see from the aforementioned articles two common models represent the power function either with PWL functions, or with a bilinear function. We will consider models using these two types of functions, but we will also consider models with more precise non-linear function $\boldsymbol{F}$ in formulation (P_{gen}). As the focus is to represent the power function, the 1-HUC considered in this paper is simplified with respect to other components. Thus, many constraints from the literature will be ignored. In particular, the spillage [28] [23] [6], which is the process of discharging water from the upstream reservoir to the downstream reservoir without going through the unit, will not be considered. The water spilled has no economic value, and spillage occurs on very rare occasions. The maximum variation of the water flow between two consecutive time periods [10], taking into account several other uses of water in the valley, will also be ignored. The start-up and shut-down costs [21] [14], making the start up and shut down

of a turbine impact the profit, are also not included in the model. We also consider a fixed turbine start-up sequence [12].

3.3 Non-linear functions considered for the 1-HUC

From generic model (P_{gen}), we want to specify functions $\boldsymbol{f}$ and $\boldsymbol{F}$. Firstly, we will focus on function $\boldsymbol{f}$ to compute the head. **Figure 2a** shows the evolution of the head depending on the volume for a realistic instance named B-T-1, described in **Appendix C**. More generally, function $\boldsymbol{f}$ has the following form:

$$h_t = \boldsymbol{f}(v_t^1) = \gamma_1 + \gamma_2 \cdot v_t^1 + \gamma_3 \cdot (v_t^1)^{\gamma_4} \qquad \forall t \leq T \tag{10}$$

where γ_i are parameters depending on the instance, and $\gamma_4 \in [0.5, 1]$, which means that this function is necessarily concave. Depending on the shape of the reservoir, the function can be quasi linear or have a very noticeable non-linearity, but always stays concave.

Secondly, we will focus on the power function $\boldsymbol{F}$. A generic hydroelectric power function from the literature [16] is described by equation (9). However, this function does not correspond to our data. Indeed, for a fixed head, the shape of the power function is known in the literature to be a non-convex and non-concave function [20] [23] [4]. The following generic function corresponds more precisely the data, and will be considered in our models instead of equation (9), with function $\boldsymbol{g}$ non-convex and non-concave.

$$p_t = \rho \cdot G \cdot h_t \cdot \boldsymbol{g}(d_t, h_t) \tag{11}$$

When it comes to the hydroelectric power function, there is no analytic function $\boldsymbol{g}$ that perfectly represents the physique. In the following we define function g as well as possible, based on data and the following information on the shape of the hydroelectric power function.

When the head is fixed, the power is the product of function $\boldsymbol{g}$ and constants, but remains non-convex and non-concave. The reason why $\boldsymbol{g}$ is non-convex and non-concave is because the function $\boldsymbol{g}$ includes the power rate of each turbine. In particular, the power rate for each turbine is concave, and the unit has N turbines which start-up in a prescribed order. Adding multiple turbines adds concavity to the resulting function. To push the analysis further, function $\boldsymbol{g}$ also has the following characteristics. For each turbine, function $\boldsymbol{g}$ is almost linear when the turbine starts, then it incurves more and more until the next turbine starts. When another turbine starts, we notice a break in the function shape. The four main characteristics of $\boldsymbol{g}$ is described in **Table 1**.

In order to be as close to the physics as possible, we define $\boldsymbol{g}$ as a piecewise non-linear function with N different five parameter logistic functions ($\mathbf{5PL}$) [17]. These $\mathbf{5PL}$ functions, described in **Appendix B**, are slightly modified to ensure they are defined for any water flow. Function $\boldsymbol{g}$ is as follows, with y a vector containing the five parameters of the $\mathbf{5PL}$:

$$\boldsymbol{g}(d_t, H) = \sum_{i=1}^{N} b_i \cdot \mathbf{5PL}_i(d_t, y)$$

C1	non-convex and non-concave
C2	locally linear when a turbine starts
C3	concave for each turbine with respect to the water flow
C4	non-differentiable points when changing turbine

Table 1: Characteristics of the power function

with the binary variable b_i equal to one if and only if the i first turbines are running, and $\mathbf{5PL}_i$ the cumulative power of the i first turbines, assuming a fixed turbine start-up sequence. The $\mathbf{5PL}_i$ functions are parameterized such that only the concave part of the $\mathbf{5PL}_i$ is considered when i turbines are running. This non-linear power function of the original model features the four main characteristics described in **Table 1**.

In the general case, function $\boldsymbol{g}$ also depends on the head. **Figure 2b** shows the evolution of function $\boldsymbol{g}$ with respect to d_t for instance B-T-1 (see **Table 9**). The functions in black are for the minimum and the maximum head, and the grey region represents the power function for the possible values of h_t. It is noticeable that function $\boldsymbol{g}$ for the maximum head is not a linear transformation of function $\boldsymbol{g}$ for the minimum head. Indeed, the shape of the function slightly changes with the head. This means that the head influences the parameters of the function $\boldsymbol{g}$. To take this effect into account, the parameters y_1 to y_5 of the $\mathbf{5PL}$ functions are linearly dependent on the head h_t.

The model (P_{gen}) with functions $\boldsymbol{g}$ and $\boldsymbol{h}$ to compute the head and the power will be referred to as the *original model*. This original model is is a non-linear model with mixed variables. Preliminary computations show that this model involves higher computing times than any other model presented, and is not practicable for most of the instances considered, even for the smallest ones. This is often the case in real world applications where the functions modeling a physical system are either too complex to be implemented or not supported by any solver.

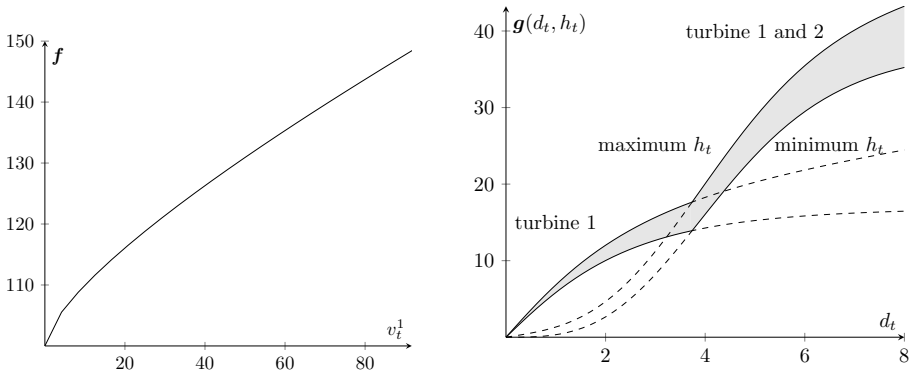
For resolution purposes, the idea is to derive more tractable models than the original model to capture the non-linearity in the power function.

4 Models for the 1-HUC

The following sections propose different models to represent the non-linearities of the 1-HUC arising from the power generation function and its characteristics. The models are also described for the fixed-head 1-HUC, which is a special case often considered.

4.1 (MI)NLP modeling elementary non-linear functions

The models in this section represent the power function of each turbine explicitly, in order to have a representation close to the physics. The downside is that for each turbine, auxiliary variables are required. Three models are presented,



(a) Example of function $\boldsymbol{f}$ (constraint (10))

(b) Example of function $\boldsymbol{g}$ for two turbines for the minimum and maximum head (constraint (11))

Figure 2: Examples of function $\boldsymbol{f}$ and $\boldsymbol{g}$

the first one features a family of polynomial functions, the second one a family of $\mathbf{5}\boldsymbol{PL}$ functions with the **max** function, and the last one a family of $\mathbf{5}\boldsymbol{PL}$ functions without the **max** function.

4.1.1 MINLP with a family of polynomial functions

The power function of one turbine for a given head has a parabolic shape. A parabolic shape can be represented with a polynomial function with degree 2. Each polynomial function represents the power generated by a turbine, plus the contribution of the previous ones, following their startup order. **Figure 3** shows an example with 3 turbines. We define the following notations:

- b_{it} : the binary variables such that $b_{it} = 1$ if we use the polynomial function of turbine i at the time period t;
- x_{kit} : the continuous variable being the coefficient of monomial d_t^k in the polynomial of turbine i at time period t;
- α_{ki} and β_{ki} : constants such that x_{kit} linearly depends on h_t with these parameters;

We introduce the following inequalities:

$$x_{kit} = \alpha_{ki} + \beta_{ki} \cdot h_t \qquad \forall k \leq 2, \forall i \leq N, \forall t \leq T \qquad (12)$$

$$\sum_{i=1}^{N} b_{it} = 1 \qquad \forall t \leq T \qquad (13)$$

$$b_{it} \in \{0, 1\} \qquad \forall i \leq N, \forall t \leq T \qquad (14)$$

$$p_t = \rho \cdot G \cdot h_t \cdot \sum_{i=1}^{N} b_{it} \cdot \sum_{k=0}^{2} x_{kit} \cdot (d_t)^k \qquad \forall t \leq T \qquad (15)$$

Inequalities (12) compute parameters x_{kit} as linearly dependent on h_t. Inequalities (13) and (14) ensure that only one of the polynomials is active for each time period. Inequalities (15) compute the power with function $\boldsymbol{g}$ represented by a family of polynomial functions.

The complete model, called $(P_{2D-poly})$ is defined by inequalities (1)-(3), (6)-(7), (10), (12)-(15). It appears that $(P_{2D-poly})$ is a non-convex MINLP as (10) and (15) are non-linear. Indeed, function $\boldsymbol{f}$ computing h_t in (10) is concave, and the polynomial functions in (15) are concave. But x_{kit} is linear with respect to h_t (12), and in (15) variable x_{kit} is multiplied by h_t. So the power function is convex with respect to h_t, as it is a growing polynomial of degree 2. Consequently, the region for the optimization is non-convex. This model represents well the power function, as it takes into account characteristics C1, C3 and C4. However, this model still has downsides, mainly the addition of auxiliary binary variables.

The polynomial functions are such that, for a given water flow, the polynomial with the highest value represents the power of the turbine. Indeed, as we maximize the profit, for a given water flow, the polynomial with the highest value, for a given d_t, will be considered for the optimal solution if energy prices Λ_t are positive. The considered instances will have $\Lambda_t \geq 0$, thus there is no need to add inequalities specifying which variable b_{it} is equal to 1 for a time period.

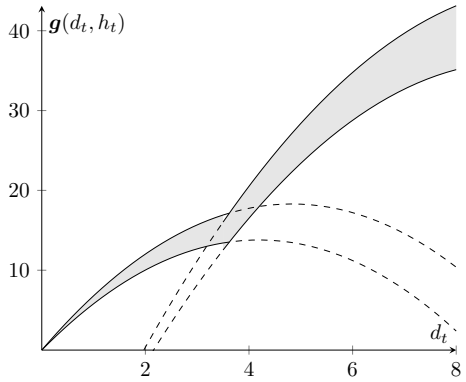


Figure 3: Function $\boldsymbol{g}$ with polynomial functions

Fixed-head 1-HUC To adapt the model ($P_{2D-poly}$) to the fixed-head 1-HUC, with head H, we introduce the following notations:

- X_{ki} : the constants coefficient of $(d_t)^k$ for the turbine i.

The power function becomes (16)

$$p_t = \rho \cdot G \cdot H \cdot \Big(\sum_{i=1}^{N} b_{it} \cdot (\sum_{k=0}^{2} X_{ki} \cdot (d_t)^k) \Big) \qquad \forall t \leq T \qquad (16)$$

Model ($P_{2D-poly}$) for the fixed-head 1-HUC contains inequalities (1)-(3), (6)-(7), (13)-(14) and (16) and is an MINLP. Indeed, we maximize the objective function, and the only non-linear functions are concave polynomials of degree 2, hence the region of optimization for the integer relaxation is convex.

4.1.2 NLP with $5PL$ functions using function max

Function $\boldsymbol{g}$ can be represented as a sum of $\mathbf{5PL}$ functions, where each $\mathbf{5PL}$ represents the power of one turbine. By summing properly parametrized functions, the sum can be a precise approximation of the physical data. **Figure 4** shows an example of the sum of $\mathbf{5PL}$ functions as a solid line, and the three separated $\mathbf{5PL}$ as dashed lines.

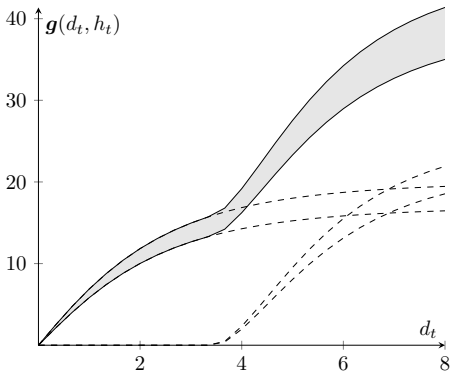


Figure 4: Function $\boldsymbol{g}$ with a sum of $\mathbf{5PL}$ functions

To represent $\boldsymbol{g}$ with as a sum of $\mathbf{5PL}$ functions, the $\mathbf{5PL}$ functions need to depend on the water flow. Besides, as function $\boldsymbol{g}$ depends on the water flow and the head h_t, the parameters of the $\mathbf{5PL}$ functions have to depend on the head. As such, these parameters denoted y_j, $j \in \{1, .., 5\}$, are decision variables. Plus, as there are multiple turbines and multiple time periods, these variables have 3 indices. To use $\mathbf{5PL}$ functions, we introduce the following notations:

- y_{jit} : continuous variable being the j^{th} parameter of the $\mathbf{5PL}$ function for the turbine i at the time period t;

- γ_{ji} : and δ_{ji} constants such that y_{jit} linearly depends on h_t with these parameters;

For this model we need the following equalities:

$$y_{jit} = \gamma_{ji} + \delta_{ji} \cdot h_t \qquad \forall j \leq 5, \forall i \leq N, \forall t \leq T \tag{17}$$

$$p_t = \rho \cdot G \cdot h_t \cdot \Big(\sum_{i=1}^{N} y_{4it} + \frac{-y_{4it}}{\Big(1 + \Big(\frac{\mathbf{max}(0, d_t - y_{1it})}{y_{3it}}\Big)^{y_{2it}}\Big)^{y_{5it}}} \Big) \qquad \forall t \leq T \tag{18}$$

Inequalities (17) set the parameters of the $\mathbf{5PL}$ functions used in function $\boldsymbol{g}$. Inequalities (18) compute the power with $\boldsymbol{g}$ as a sum of $\mathbf{5PL}$ functions. As mentioned, a $\mathbf{5PL}$ function is not defined if $x < y_1$, which means for the 1-HUC when $d_t < y_{1it}$. This is why we introduce function **max** in equalities (18).

The model ($P_{5PL-max}$) includes inequalities (1)-(3), (6)-(7), (10) and (17)-(18). It is a non-convex non-concave NLP, and the non-linearity takes into account characteristics C1 and C2. As this model contains a **max** function, it is not supported by some global optimization solvers.

Fixed-head 1-HUC To adapt the model ($P_{5PL-max}$) to the fixed-head 1-HUC, with head H, we introduce the following notations:

- $Y_{1i}, .., Y_{5i}$: the 5 parameters for the $\mathbf{5PL}$ function of the turbine i.

The power is computed by equalities (19).

$$p_t = \rho \cdot G \cdot H \cdot \Big(\sum_{i=1}^{N} Y_{4i} + \frac{-Y_{4i}}{\Big(1 + \Big(\frac{\mathbf{max}(0, d_t - Y_{1i})}{Y_{3i}}\Big)^{Y_{2i}}\Big)^{Y_{5i}}} \Big) \qquad \forall t \leq T \tag{19}$$

Model ($P_{5PL-max}$) for the fixed-head 1-HUC contains inequalities (1)-(3), (6)-(7) and (19).

4.1.3 MINLP with $5PL$ functions using auxiliary variables

This model is a variation of ($P_{5PL-max}$), where the **max** is linearized by adding linear inequalities and auxiliary variables. With the addition of binary variables a_{it}, we introduce the following set of inequalities:

$$d_t - y_{1it} \leq u_{it} \leq d_t - y_{1it} + (1 - a_{it}) \cdot (\overline{D} - \underline{Y}_{1it}) \qquad \forall i \leq N, \forall t \leq T \tag{20}$$

$$0 \leq u_{it} \leq a_{it} \cdot (\overline{D} - \underline{Y}_{1it}) \qquad \forall i \leq N, \forall t \leq T \tag{21}$$

$$a_{it} \in \{0, 1\} \qquad \forall i \leq N, \forall t \leq T \tag{22}$$

$$p_t = \rho \cdot G \cdot h_t \cdot \Big(\sum_{i=1}^{N} y_{4it} + \frac{-y_{4it}}{\Big(1 + \Big(\frac{u_{it}}{y_{3it}}\Big)^{y_{2it}}\Big)^{y_{5it}}} \Big) \qquad \forall t \leq T \tag{23}$$

Set of inequalities (20)-(22) ensure $u_{it} = \mathbf{max}(0, d_t - y_{1it})$. Inequalities (23) compute the power in the same manner as inequalities (18), but using u_{it}.

The model $(P_{5PL-bin})$ contains inequalities (1)-(3), (6)-(7), (10), (17) and (20)-(23). Unlike the NLP model $(P_{5PL-max})$, model $(P_{5PL-bin})$ is an MINLP, as it requires auxiliary binary variable a_{it}. Model $(P_{5PL-bin})$ can be solved by many more MINLP solvers, as function **max** has been linearized. The non-linearity is the same for $(P_{5PL-max})$ and $(P_{5PL-bin})$, and both models take into account characteristics C1 and C2. Note that the model (P_{gen}), with the piecewise non-linear function with $\mathbf{5}\boldsymbol{PL}$, is also an MINLP. The difference is that the binary variables are not the same as the ones in $(P_{5PL-bin})$. Indeed, the binary variables of $(P_{5PL-bin})$ only acts in order to linearize the function **max**, while in (P_{gen}) they are decision variables.

Fixed-head 1-HUC To adapt the model $(P_{5PL-bin})$ to the fixed-head 1-HUC, with head H, we introduce the following notations:

- Y_{ji} : constants for the parameter j for the $\mathbf{5}\boldsymbol{PL}$ function of the turbine i;
- u_{it} : the variable such that $u_{it} = \mathbf{max}(0, d_t - Y_{1i})$.

To ensure the behaviour of variable u_{it}, we add the set of inequalities (24)-(27), defined as (20)-(23), where variables y are replaced by constants Y. Model $(P_{5PL-bin})$ for the fixed-head 1-HUC contains inequalities (1)-(3), (6)-(7) and (24)-(27).

4.2 (MI)NLP modeling an aggregated non-linear function

The models introduced in this section represent all turbines as an aggregated function. The principle is to consider a single function to represent the whole power function, instead of a family or a sum of elementary functions. A single function being less precise, the expected benefit is a quick resolution by MINLP tools, as few additional variables and inequalities are required. The functions we propose are the following: a polynomial function, a bilinear function, and a finite set of operating flows.

4.2.1 NLP with a high degree polynomial function

A model using an aggregated function that represents well the physics is obtained by using a single polynomial function as function $\boldsymbol{g}$. **Figure 5** shows an example of an 8^{th} degree polynomial function for an instance with two turbines. As $\boldsymbol{g}$ depends on the head h_t, the coefficients of the polynomial are linearly dependent on h_t. We introduce the following notation

- Q : the degree of the polynomial, with $Q = 4N$, where N denotes the number of turbines;
- z_{qt} : the continuous variable being the coefficient of monomial d_t^q in the polynomial function at time period t;

- η_q : and θq constants such that z_{qt} linearly depends on h_t with these parameters;

For this model, we need the following inequalities:

$$z_{qt} = \eta_q + \theta q \cdot h_t \qquad \forall q \leq Q, \forall t \leq T \tag{28}$$

$$p_t = \rho \cdot G \cdot h_t \cdot \Big(\sum_{q=0}^{Q} z_{qt} \cdot (d_t)^q \Big) \qquad \forall t \leq T \tag{29}$$

Inequalities (28) set the parameters of the polynomial function. Inequalities (29) compute the power with $\boldsymbol{g}$ as a polynomial function.

The model, called $(P_{HD-poly})$ includes inequalities (1)-(3), (6)-(7), (10) and (28)-(29). It is an NLP featuring characteristic C1 as it is non-convex and non-concave. The benefits compared to $(P_{2D-poly})$ is that $(P_{HD-poly})$ only considers one single polynomial function. This means that no auxiliary binary variables are required. The downside of $(P_{HD-poly})$ is that high degree polynomials (8 for two turbines, 20 for five turbines) can induce large approximation errors. If the water flow can fluctuate between 0 and 100, then it means that the solver might need to compute numbers such as 0.1^8 or 100^8, which are either too small or too large numbers for solvers' precision. Moreover, computational errors can have a dramatic impact for the HUC, as an error for a time period will cumulate and carry over all future time periods [31].

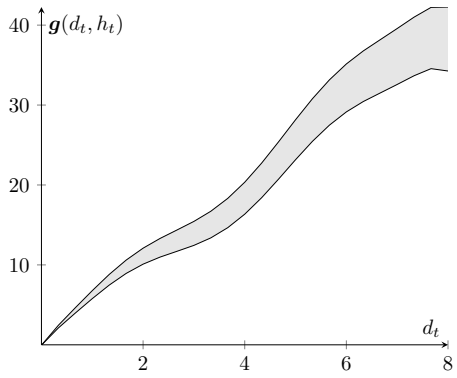


Figure 5: Function $\boldsymbol{g}$ with a single polynomial function

Fixed-head 1-HUC To adapt the model $(P_{HD-poly})$ to the fixed-head 1-HUC, with head H, we introduce the following notations:

- Z_q : the coefficients for the degree q of the polynomial function.

The power function becomes:

$$p_t = \rho \cdot G \cdot H \cdot \Big(\sum_{q=0}^{Q} Z_q \cdot (d_t)^q \Big) \qquad \forall t \leq T \tag{30}$$

The model ($P_{HD-poly}$) for these special instances contains inequalities (1)-(3), (6)-(7) and (30).

4.2.2 NLP with a bilinear function

A type of model often described in the literature to solve the HUC as an MINLP is a bilinear model [23], [6]. The power is linear with respect to the water flow, and to the head. In the model (P_{gen}), the power is already linear with respect to the head. We need to make it also linear with respect to the water flow. To do so, we introduce the following notations:

- μ and ν: constants such that the power is linearly dependant on the water flow.

We adapt the power function as follows:

$$p_t = \rho \cdot G \cdot h_t \cdot (\mu + \nu \cdot d_t) \qquad \forall t \leq T \tag{31}$$

Inequalities (31) compute the power linearly dependent of the head h_t and the water flow d_t.

The model (P_{bilin}) contains inequalities (1)-(3), (6)-(7), (10) and (31). As mentioned, inequality (31) is a bilinear function of h_t and d_t. A bilinear function is non-convex non-concave. However, the constraint are such that $h_t \geq 0$ and $d_t \geq 0$, in which case a bilinear function is concave. As we maximize the objective function, and both h_t and p_t are computed with a concave function, the region for the optimization is convex. Unlike ($P_{2D-poly}$), ($P_{HD-poly}$), ($P_{5PL-max}$) or ($P_{5PL-bin}$), (P_{bilin}) requires no additional variables. This makes this model a potential candidate to solve quickly the problem. The downside is that this model has the roughest approximation of all models. Indeed, the bilinear function features none of the non-linear characteristics C1, C2, C3 or C4. Even MILP models such as PWL models (**Section 4.3**) might have a better precision.

Fixed-head 1-HUC When we adapt the model (P_{bilin}) to the fixed-head 1-HUC, with head H, the model becomes a linear model, where the power is a linear function of the water flow. To do so, we simply adapt the power function as follows:

$$p_t = \rho \cdot G \cdot H \cdot (\mu + \nu \cdot d_t) \qquad \forall t \leq T \tag{32}$$

The model (P_{bilin}) for the fixed-head 1-HUC contains inequalities (1)-(3), (6)-(7) and (32), which yields an LP.

4.2.3 MINLP with a discrete set of decisions

Having a discrete set of decisions for the 1-HUC means that only a given number, say L, operating flows are authorized. These operating flows are specifically chosen where the power production is the most profitable, and usually are in

the concave parts of the original power function in (P_{gen}). We introduce the following notations:

- D_l : the constant being the l^{th} operating flow;
- o_{lt} : the binary variable such that $o_{lt} = 1$ if we use the l^{th} operating flow at time period t.

We need the following inequalities: We consider a model with disjunctive constraints between the operating flows. As such, we need the following inequalities.

$$\sum_{l=1}^{L} o_{lt} \leq 1 \qquad \forall t \leq T \tag{33}$$

$$v_{t'}^1 = V_0^1 + \sum_{t=1}^{t'} \left(A_t^1 - (\sum_{l=1}^{L} o_{lt} \cdot D_l) \cdot \Delta \right) \qquad \forall t' \leq T \tag{34}$$

$$v_{t'}^2 = V_0^2 + \sum_{t=1}^{t'} \left(A_t^2 + (\sum_{l=1}^{L} o_{lt} \cdot D_l) \cdot \Delta \right) \qquad \forall t' \leq T \tag{35}$$

$$p_t = \rho \cdot G \cdot h_t \cdot \boldsymbol{g}(\sum_{l=1}^{L} o_{lt} \cdot D_l, h_t) \qquad \forall t \leq T \tag{36}$$

Inequalities (33) ensure that only one operating flow can be active at each time period. The set of equations (34)-(36) corresponds to equations (2), (3) and (5) from (P_{gen}), with operating flows instead of the water flow d_t.

This leads to a new generic model (P_{op}) containing inequalities (1), (6), (10) and (33)-(36). Function $\boldsymbol{g}$ in (36) can be any of the previously described function for models $(P_{HD-poly})$, $(P_{2D-poly})$, $(P_{5PL-max})$, $(P_{5PL-bin})$ or (P_{bilin}). Because we have a finite set of operating flows, function $\boldsymbol{g}$ for (P_{op}) will feature none of the characteristics C1 to C4, regardless of the function considered. Model (P_{op}) can be beneficial as the space of solution is drastically smaller, but does not offer as much freedom, in particular when target volumes occur. As the operating flows are amongst the most profitable ones, the solution might still be close to the optimal solution. The downside of this model is that target volumes can be unreachable with the set of operating flows, thus leading to infeasibility.

Fixed-head 1-HUC When we adapt the model (P_{op}) for the fixed-head 1-HUC, with head H, the model becomes an MILP model. In this case the power only depends on the water flow, thus there is a finite set of possible powers. As such, the model becomes an MILP as we have to chose a pair (operating flow, power produced) amongst a list of pairs at each time period. We introduce the following notations:

- P_l : the constant being the power generated for the l^{th} operating flow.

The model is very similar to (P_{op}) for the general 1-HUC, but we compute the power differently as we use constants P_l:

$$p_t = \sum_{l=1}^{L} o_{lt} \cdot P_l \qquad \forall t \leq T \tag{37}$$

The model (P_{op}) adapted for the fixed-head 1-HUC contains inequalities (1), (6), (33)-(35) and (37). We can notice that this model contains very few variables, as only the decision variables o_{lt} are required.

4.3 MILP modeling a PWL functions

Using a PWL approximation is a common practice when modeling a non-linear problem. It is indeed an efficient way to obtain an MILP. The aim of comparing a PWL model to the previously described non-linear models is mostly to compare the precision in terms of solution value. As such, we will consider a standard PWL formulation [8] [15], more precisely the convex combination formulation. There exist much more efficient formulations, e.g. the logarithmic formulation in [37], but they will not be considered as it will not impact the value of the solution, but only the computing time.

A generic way to obtain a two dimensional PWL function is to use the one-dimensional method described in [10]. It is a generalization of the convex combination formulation [15]. The method described to approximate a non-linear function $\boldsymbol{f}(x, y)$ is as follows. We fix I variables on the x axis, $(\widetilde{x}_1, .., \widetilde{x}_I)$, and J variables on the y axis $(\widetilde{y}_1, .., \widetilde{y}_J)$. For each $\widetilde{y}_j$, we approximate $f(x, \widetilde{y}_j)$ with a PWL function $\boldsymbol{l}(x, \widetilde{y}_j)$, where each $\widetilde{x}_i$, $i \leq I$ acts like a break point. It means that piece i of $\boldsymbol{l}(x, y_j)$ is a linear function between $\widetilde{x}_i$ and $\widetilde{x}_{i+1}$. We obtain then J PWL functions with $I - 1$ pieces. The value for $\boldsymbol{l}(x, y)$, $y \in [\widetilde{y}_j, \widetilde{y}_{j+1}]$, is approximated by $\boldsymbol{l}(x, \widetilde{y}_j)$. For the 1-HUC, we will approximate the power function with respect to the water flow d_t for a set of fixed volumes $\widetilde{y}_j$, $j \in 1, ..., J$.

In this model, we will aggregate both non-linear functions $\boldsymbol{f}$ and $\boldsymbol{g}$ as a a unique function to represent the power. To do so, it is possible to replace h_t by $f(v_t^1)$ in inequalities (5) from ($\mathcal{P}_{gen}$). Thus, the power is defined as follows, and we only need to approximate one two-dimensional non-linear function for the whole model:

$$p_t = \boldsymbol{F}(d_t, v_t^1) = \rho \cdot G \cdot \boldsymbol{f}(v_t^1) \cdot \boldsymbol{g}(d_t, \boldsymbol{f}(v_t^1))$$

To use the PWL approximation, we introduce the following notations:

- $\boldsymbol{l}(v_t^1, d_t)$: the PWL approximation of $\boldsymbol{F}(v_t^1, d_t)$;
- J : the number of PWL functions allocated on the volume axis;
- I : the number of breakpoints on the water flow axis;
- $\tilde{v}_j^1$: the volume corresponding to the j^{th} one dimensional PWL function.

- $\tilde{d}_i$: the breakpoint i for all J PWL functions;

And we include the following variables:

- $l_{j,t}$: the value $\boldsymbol{l}(\tilde{v}_j^1, d_t)$ of the PWL function j at time period t;
- r_{it} : binary variables such that $r_{it} = 1$ if d_t is located on the interval $[\tilde{d}_i, \tilde{d}_{i+1}]$;
- w_{it} : continuous variables such that d_t is the convex combination $w_{it}\tilde{d}_i + w_{i+1t}\tilde{d}_{i+1}$;
- s_{jt} : binary variables such that $s_{jt} = 1$ if v_t^1 is located in the interval $[\tilde{v}_j^1, \tilde{v}_{j+1}^1]$.

The PWL formulation requires the following inequalities:

$$\sum_{i=1}^{I} r_{it} = 1 \qquad \forall t \leq T \qquad (38)$$

$$w_{it} \leq r_{i-1t} + r_{it} \qquad \forall i \leq I, \forall t \leq T \qquad (39)$$

$$\sum_{i=1}^{I} w_{it} = 1 \qquad \forall t \leq T \qquad (40)$$

$$d_t = \sum_{i=1}^{I} w_{it} \cdot \tilde{d}_i \qquad \forall t \leq T \qquad (41)$$

$$l_{j,t} = \sum_{i=1}^{I} w_{it} \cdot \boldsymbol{F}(\tilde{v}_j^1, \tilde{d}_i) \qquad \forall j \leq J, \forall t \leq T \qquad (42)$$

$$r_{it} \in \{0, 1\} \qquad \forall i \leq I, \forall t \leq T \qquad (43)$$

$$0 \leq w_{it} \leq 1 \qquad \forall i \leq I, \forall t \leq T \qquad (44)$$

$$\sum_{j=1}^{J} s_{jt} \cdot \tilde{v}_j^1 \leq v_t^1 \leq \sum_{j=1}^{J} s_{jt} \cdot \tilde{v}_{j+1}^1 \qquad \forall t \leq T \qquad (45)$$

$$\sum_{j=1}^{J} s_{jt} = 1 \qquad \forall t \leq T \qquad (46)$$

$$l_{j,t} - \overline{P}_t \cdot (1 - s_{jt}) \leq p_t \leq l_{j,t} + \overline{P}_t \cdot (1 - s_{jt}) \qquad \forall j \leq J, \forall t \leq T \qquad (47)$$

$$s_{jt} \in \{0, 1\} \qquad \forall j \leq J, \forall t \leq T \qquad (48)$$

Inequalities (38)-(44) are the standard convex combination formulation for a one-dimensional PWL function, applied to approximate function $\boldsymbol{F}$ for each given volume. These inequalities ensure that $l_{j,t}$ is the value, at time period t of the PWL function approximating $\boldsymbol{F}$ for for volume $\tilde{v}_j^1$. Inequalities (38) express that exactly one variable r_{it} is equal to one at time period t, meaning that we consider one piece of the PWL function at time period t. Inequalities (39) allows

the weight r_{it} of a breakpoint to be greater than zero only if one for the two surrounding pieces is considered at time period t. Inequalities $(40)-(41)$ ensure that d_t is the convex combination of the $\tilde{d}_i$ with the weights r_{it} at time period t. Inequalities (42) compute the PWL approximation of the power function.

We have described the inequalities to obtain $l_{t,j}$ the approximated value of $\boldsymbol{F}$ at time period t using univariate PWL functions. Now we need inequalities (45)-(48) in order to compute the power from the value $l_{t,j}$ for the fixed volume $\tilde{v}_j^1$, with $\tilde{v}_j^1 \leq v_t^1 \leq \tilde{v}_{j+1}^1$. Inequalities (45)-(47) ensure $s_{j,t} = 1$ if $v_t^1 \in [\tilde{v}_j^1; \tilde{v}_{j+1}^1]$, and exactly one variable $s_{j,t}$ is equal to 1. Variable $s_{j,t}$ then indicates which PWL function should be considered depending on the volume. Inequalities (47) ensure $p_t = l_{j,t}$ if $s_{j,t} = 1$, or give trivial bounds if $s_{j,t} = 0$.

The MILP model (P_{pwl}) contains inequalities (1)-(3), (6)-(7) and (38)-(48). The consequences of this model being an MILP are twofold. On the one hand, it can be solved with powerful MILP tools. On the other hand it includes a lot of auxiliary variables and inequalities, and it does not include any of the non-linear characteristics of the power function.

Fixed-head 1-HUC When we adapt the model (P_{pwl}) to the fixed-head 1-HUC, with head H, we approximate a one-dimensional power function. To do so, we use the convex combination formulation [15], which is the formulation generalized for formulation (P_{pwl}) in the general case. The convex combination formulation adapted for the 1-HUC requires inequalities (38)-(41) and (42)-(44). We compute the power as follows

$$p_t = \sum_{i=1}^{I} w_{it} \cdot \rho \cdot G \cdot H \cdot \boldsymbol{g}(\tilde{d}_i, H) \qquad \forall t \leq T \qquad (49)$$

Thus the model (P_{pwl}) for these special instances contains inequalities (1)-(3), (6)-(7), (38)-(41) and (43)-(44) and (49)

4.4 Summary of models and non-linear functions

We have described a total of fourteen different models. Most of the models share the same set of constraints. We define constraint sets S1=(1)-(3), (6)-(7), (11) and S2=(1)-(3), (6)-(7). **Table 2** summarizes all models with their constraints. The difference between these models is the representation of the power function.

Table 3 shows the characteristics and the type of program for each model. The convexity and the linearity of a model do not take into account the integer variables. From a theoretical point of view, none of the presented models perfectly fits the power function of the original model. Indeed, none of the models features all four non-linear characteristics of the power functions. However, it will be shown in the numerical experiments that some models allow for very small approximation errors, while other models, with simpler non-linear expressions, lead to shorter computing times.

It is also possible to compare the models, on the basis of the difficulty for the solvers to manage their non-linear expression. A way to measure this is to

Model	General cases inequalities	Fixed-head cases inequalities
$(P_{2D-poly})$	S1,(12)-(15)	S2,(13)-(14), (16)
$(P_{5PL-max})$	S1,(17)-(18)	S2,(19)
$(P_{5PL-bin})$	S1,(17),(20)-(23)	S2,(24)-(27)
$(P_{HD-poly})$	S1,(28)-(29)	S2,(30)
(P_{bilin})	S1,(31)	S2,(32)
(P_{op})	(1), (6),(10), (33)-(36)	(1), (6), (33)-(35), (37)
(P_{pwl})	S2,(38)-(48)	S2,(38)-(41), (42)-(44), (49)

Table 2: Summary of the proposed models

	1-HUC		Fixed-head 1-HUC		Characteristics			
Model	Type	Convexity	Type	Convexity	C1	C2	C3	C4
$(P_{2D-poly})$	MINLP	non-convex	MINLP	convex	✓	✗	✓	✓
$(P_{5PL-max})$	NLP	non-convex	NLP	non-convex	✓	✓	✗	✗
$(P_{5PL-bin})$	MINLP	non-convex	MINLP	non-convex	✓	✓	✗	✗
$(P_{HD-poly})$	NLP	non-convex	NLP	non-convex	✓	✗	✓	✗
(P_{bilin})	NLP	convex	LP	linear	✗	✗	✗	✗
(P_{op})	MINLP	non-convex	MILP	linear	✗	✗	✗	✗
(P_{pwl})	MILP	linear	MILP	linear	✗	✗	✗	✗

Table 3: Comparison of the models non-linear characteristics

compare the size of the reformulation binary tree for the non-linear expressions as in [34]. Following this metric, $\mathbf{5}\boldsymbol{PL}$ functions are by far the most difficult functions, followed by high degree polynomials, two degree polynomials, bilinear functions and linear functions.

5 Numerical experiments

The tests are performed via Neos Server [9] using the following five MINLP solvers: ANTIGONE, BARON, COUENNE, LINDOGlobal, SCIP, along with the MILP solver CPLEX. For MINLP solvers, the GAMS format is used for input files, while for the MILP solver, the LP format is used. All experiments are performed on Neos Server machine prod-exec-7 (a 2x Intel Xeon Gold 5218 @ 2.3GHz processor with 384 GB of RAM), using a single thread. The computing time limit is set to 10800 seconds.

5.1 Modeling choices

The parameters of the power functions featured in the different models are obtained by fitting them to the power function of the original model. The fitting is done via Scipy's curve_fit function[1], using a non-linear least squares

[1] `https://docs.scipy.org/doc/scipy/reference/generated/scipy.optimize.curve_fit.html`, accessed: 2023-01-09

method. Unfortunatly, no a priori precision is given for the resulting function with respect to the data. Recall that the purpose of this work is to study and analyse various approximations of the power function. Thus the parameters of the head functions of all proposed models are the ones of the original model.

For model (P_{pwl}) we also want to compare the impact of the number of linear pieces, therefore we will define three variants of (P_{pwl}): (P^1_{pwl}), (P^2_{pwl}) and (P^3_{pwl}), respectively with 5, 20 and 100 breakpoints for each axis. For every variant, the breakpoints are defined as equidistant instead of being tailored to each instance. Results show that model (P_{pwl}) is not significantly penalized in terms of approximation error compared to the other models, with equidistant breakpoints.

For model (P_{op}), a discrete set of decision variables is to be chosen. For the considered instances, we define $\boldsymbol{g}$ as the non-linear function of model $(P_{5PL-bin})$, and we consider 5 operating flows per turbine. We will not consider models with more operating flows, as the model contains $\mathbf{5}\boldsymbol{PL}$ functions that are already difficult to handle. Additional operating flows would make the model irrelevant as it would become too hard to solve.

All models contain variables that are subject to an equality constraint, but there are no explicit constraint to bound the value of these variables. However, when using global solvers, it is a good practice to bound every variable. Hence, for the experiments, every variable has an upper and lower bound, even if these bounds are trivially set through the equality constraints.

5.2 Instances

All the instances on which the models and solvers are tested are described in detail in **Appendix C**. Different characteristics of the instances are changed from an instance to another. The varying characteristics and the corresponding 1-HUC parameters are as follows. The *size of the instance* varies with the number of time periods. *Equality constraints* appear as soon as target volumes are accounted for in the instance. Two characteristics of the non-linear function can be changed: the *number of inflection points* and the *degree of non-linearity*. These characteristics are respectively linked to the number of turbines, and to when the transition from a turbine to another occurs when increasing or decreasing the water flow. The last characteristic is the *sensitivity of the decision variables*, which measures how much the decision can affect the dynamical behavior of the physical system. For the 1-HUC, the smaller the water flows relative to the absolute volume, the less the volumes change over the time periods. The sensitivity S can be computed as follows for the 1-HUC:

$$S = \frac{\overline{D} \cdot T - \underline{D} \cdot T}{\min(\overline{V}^1_T - \underline{V}^1_T, \overline{V}^2_T - \underline{V}^2_T)}$$

For instance, let an instance of the 1-HUC be with $\overline{D} = 50$, $\underline{D} = 0$, $T = 10$, $\overline{V}^1_T = \overline{V}^2_T = 1000$ and $\underline{V}^1_T = \underline{V}^2_T = 0$, then $S = 500/1000 = 0.5$. A similar instance with $\overline{V}^1_T = \overline{V}^2_T = 1000$ would have $S = 500/10000 = 0.05$.

Table 9 in **Appendix C** summarizes the instances and their characteristics.

5.3 Terminology, notations and metrics

In this section, additional terminology is introduced to compare the different models on the considered instances. For the comparisons, we also define the metrics used and their notation.

A *configuration* is defined as a pair (instance, model). For all models, a configuration is solved to optimality when the optimality gap between the primal and dual values is below 0.1%. Note that the optimality gap is not computed in the exact same manner for every solver, but remains very similar. The *optimal solution of a configuration* is the solution when the configuration is solved to optimality. The value of a configuration is the value of the optimal solution. The *recalculated value of a solution* is the value of the solution, evaluated with the non-linear functions of the original model. The *recalculated value of a configuration* is the recalculated value of the optimal solution of the configuration. A configuration is *solvable* by a solver if the solver supports the model. A configuration is *solved* if the it is solved to optimality with at least one solver. A configuration is *feasible* with a solver when it is not solved to optimality, but a solution is found. A configuration is *infeasible* with a solver if the solver proves the configuration to have no feasible solution.

The metrics used to compare the models and the solvers are as follows. The *computing time* (CT) of a configuration is the time required to return the optimal solution. The *approximation error* (AE) of a configuration is the relative difference between the value of the optimal solution of the configuration, and the recalculated value of the configuration. The *distance to the best recalculated value* (DB) of a configuration is the relative difference between the recalculated value of the configuration, and the highest recalculated value of all configurations with the same instance.

As specified, configurations are solved with several solvers. We define the *virtual best solver* (VBS) [18] of a given configuration as the solver that requires minimal CT to solve the configuration. Results show that the AE (resp. the DB) of a configuration is the same for every solver. Thus, for our results, the VBS is the solver that has the configuration solved to optimality in minimal CT. For our analysis we use the metrics of the configurations with their VBS. All figures and tables for the results are with the VBS, except for **Tables 5** and **6** that display the results for each solver.

5.4 Model comparison

As aforementioned, the 1-HUC and its simplification, the fixed-head 1-HUC, are considered. To observe the effect of this simplification, results are presented for both the 1-HUC and the fixed-head 1-HUC.

Note that some configurations are not solvable with every solver. Indeed, model ($P_{5PL-max}$) is only supported by LINDOGlobal and SCIP. Besides, none

of the configurations with $(P_{HD-poly})$ returns a feasible solution. It follows that results related to model $(P_{HD-poly})$ are not enclosed.

Figure 6 shows on the y-axis the proportion of configurations solved with their VBS, under a CT threshold given on the x-axis. In a similar manner, **Figure 7** shows on the y-axis the proportion of configurations solved with their VBS, but this time under a given AE threshold on the x-axis. Analogously, **Figure 8** shows on the y-axis the proportion of configurations solved with their VBS, under a given DB threshold on the x-axis. For these three figures, the configurations are color-coded depending on the model of the configuration.

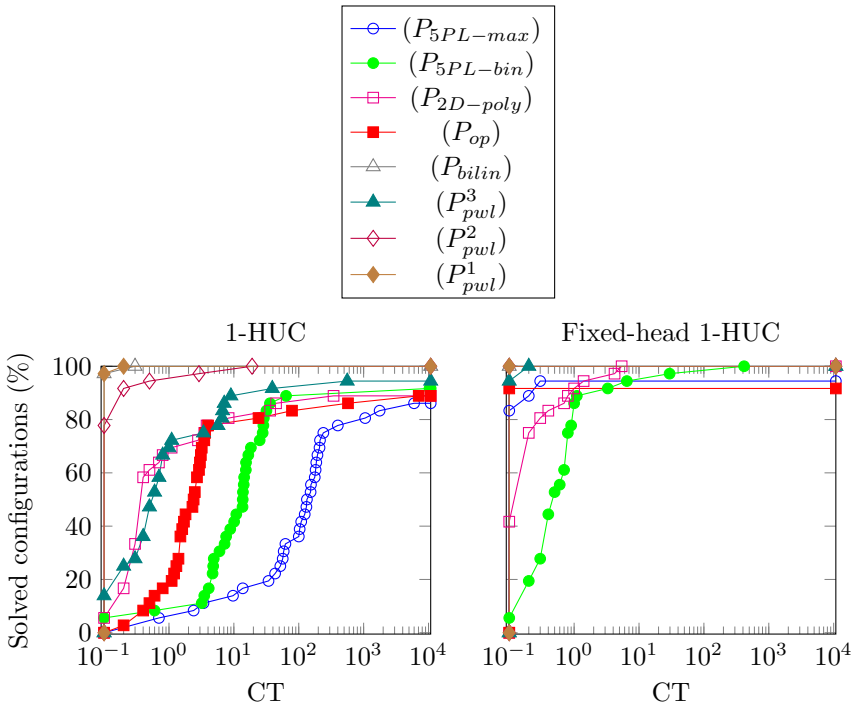


Figure 6: Proportion of configurations solved with their VBS where the CT is under a CT threshold

For any model the 1-HUC requires more CT (**Figure 6**) and leads to smaller AE (**Figure 7**) than the fixed-head 1-HUC.

We notice from **Figure 6** and **Figure 7** that for the 1-HUC, a model is a trade-off between CT and AE, whereas for the fixed-head 1-HUC, every model has similar AE, except for (P_{bilin}), which yields a poor approximation. When considering the fixed-head 1-HUC, it becomes counterproductive to use a very sophisticated non-linear model, as the AE induced by a fixed head will be largely independent of the chosen model.

We now evaluate the economic quality of the solution obtained by comparing the DB (**Figure 8**). Most configurations yielding the smallest DB are with the

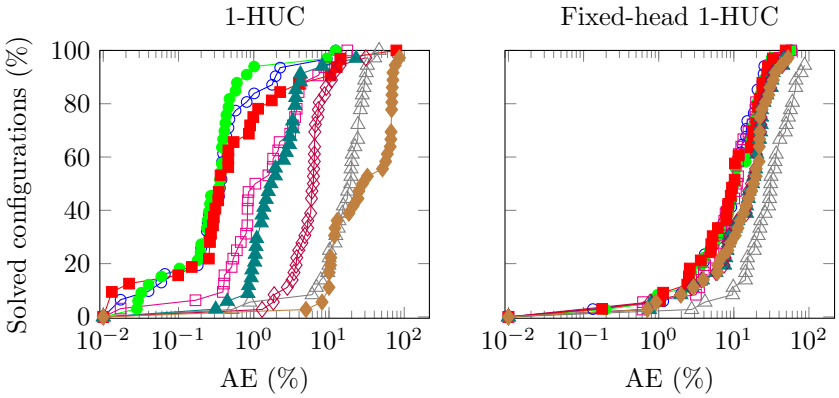


Figure 7: Proportion of configurations solved with their VBS where the AE is under an AE threshold

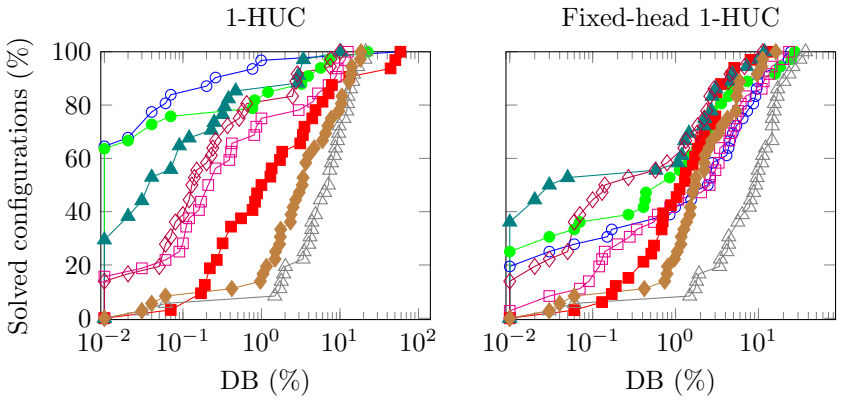


Figure 8: Proportion of configurations solved with their VBS where the DB is under a DB threshold

1-HUC. For the fixed-head 1-HUC, the optimal solution of a configuration may be far from optimal for (P_{gen}), due to the generally large AE. However it must be noted that there are cases where the DB for the fixed-head 1-HUC is small. This can happen in two cases: when only configurations with non-precise models ((P^2_{pwl}), (P^3_{pwl}) and (P_{bilin})) are solved for the 1-HUC, or for configurations relative to instances with low sensitivity of decision variables. In the latter case, the AE of the fixed-head 1-HUC is similar to the AE of the 1-HUC.

The trend is for models with small (resp. high) AE (**Figure 7**) to also have small (resp. high) DB (**Figure 8**). However we notice an exception for (P_{op}). Indeed, model (P_{op}) has a similar AE as ($P_{5PL-max}$) and ($P_{5PL-bin}$) (**Figure 7**). However the DB for (P_{op}) is much higher than for ($P_{5PL-max}$) and ($P_{5PL-bin}$) (**Figure 8**), which means that its solutions are of lesser economic

quality. This is because for (P_{op}), there is a finite set of operating flows, and every solution must have water flows within this set. It is possible that the optimal solutions of configurations with (P_{op}) are far from the optimal once the water flows are not restricted to the finite set. Note that (P_{op}) is the only model that yields a DB above 30% for the 1-HUC.

To summarize the results, **Figure 9** shows a bargraph which represents two categories of results: the proportion of configurations solved with their VBS for each model, and the proportion of these configurations where the DB is the smallest (resp. the second smallest, third smallest) compared to other configurations with the same instance. Note that if two configurations with the same instance lead to the smallest (resp. second smallest, third smallest) DB, they are both counted as leading to the smallest (resp. second smallest, third smallest) DB. These results are distinguished for both the 1-HUC and the fixed-head 1-HUC.

Besides models (P^1_{pwl}), (P^2_{pwl}) and (P_{bilin}) that solve every configuration for the 1-HUC and for the fixed-head 1-HUC, the proportion of configurations solved is larger for the fixed-head 1-HUC than for the 1-HUC. Moreover in the fixed-head case, configurations are solved for every model but (P_{op}), due to target volumes that may not be reachable with the finite set of operating flows.

Note that for the 1-HUC, models $(P_{5PL-max})$ and $(P_{5PL-bin})$ have for most configurations a DB that is amongst the top three smallest. However for the fixed-head 1-HUC, these models lose this property with respect to their DB, and models (P^2_{pwl}) and (P^3_{pwl}) often lead to the smallest DB. This supports the point that in the case of the fixed-head 1-HUC, it is unnecessary to have a sophisticated non-linear model.

Another interesting metric is the proportion of infeasible configurations with their VBS. **Table 4** shows, for each model, the proportion of solved (%solved), feasible (%feasible), and infeasible (%infeasible) configurations with their VBS. Note that there is no case where the status is undefined: for every configuration, either a feasible solution is found, or the infeasibility is proven within three hours.

The infeasible status can occur when the configuration indeed does not have any feasible solution. For example a configuration with model (P_{op}) and an instance with target volumes is infeasible with any solver if the target volumes are not reachable with the finite set of operating flow. This is why 8.33% of configurations with model (P_{op}) are infeasible with their VBS. However, infeasibility can also occur when a configuration has a feasible solution. As $(P_{2D-poly})$ has continuous variables for the water flows d_t, and a power function defined on the domain of variables, there should be at least one solution for every configuration. Nevertheless, 11.11% of configurations with model $(P_{2D-poly})$ are infeasible with their VBS for the 1-HUC. This happens for configurations with model $(P_{2D-poly})$ featuring high degree of non-linearity: every non-linear solver returns the infeasible status for these configurations.

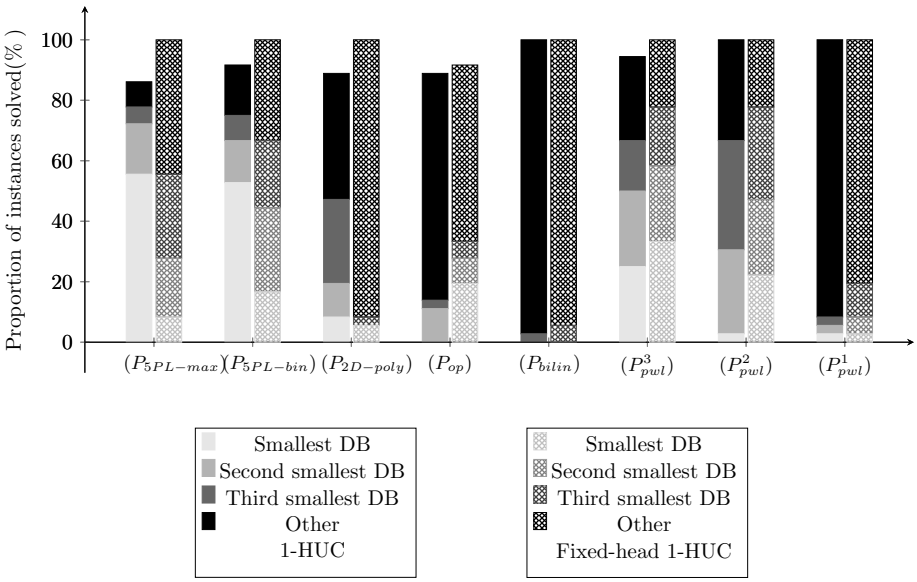


Figure 9: Proportion of configurations for each model solved by their VBS

	1-HUC			Fixed-head 1-HUC		
Model	%solved	%feasible	%infeasible	%solved	%feasible	%infeasible
$(P_{5PL-max})$	86.1	13.9	0.0	100.0	0.0	0.0
$(P_{5PL-bin})$	91.7	8.3	0.0	100.0	0.0	0.0
$(P_{2D-poly})$	88.9	0.0	11.1	100.0	0.0	0.0
(P_{op})	88.9	2.8	8.3	91.7	0.0	8.3
(P_{bilin})	100.0	0.0	0.0	100.0	0.0	0.0
(P^3_{pwl})	94.4	5.6	0.0	100.0	0.0	0.0
(P^2_{pwl})	100.0	0.0	0.0	100.0	0.0	0.0
(P^1_{pwl})	100.0	0.0	0.0	100.0	0.0	0.0

Table 4: Proportion of solution status returned for the configurations for each model by their VBS

5.5 Solver comparison

All reported results so far are with the VBS, whereas in this section the results show the behaviour of each solver independently. **Table 5** and **Table 6** show for each model the proportion of configurations solved with each solver, and the average CT, respectively for the 1-HUC and for the fixed-head 1-HUC. As aforementioned, a solved configuration has, for every solver, the same AE, only the proportion of configurations solved and the CT can change from a solver to another. In the tables, the following notations are used:

- %S: proportion of configurations solved
- avg-CT: average CT

- NS: model not supported by the solver
- NR: model supported by the solver, but experiments are not reported

For a model, a solver dominates another solver if it has a higher %S, and a smaller avg-CT. For each model, the smallest avg-CT and the highest %S are emphasized in bold. If for a model, the two metrics are in bold for a solver, then it dominates every other solver for the model.

The results confirm that the performance of a solver highly depends on the model. For instance, when the 1-HUC is considered (**Table 5**), solver SCIP dominates solver ANTIGONE for model ($P_{5PL-bin}$), but ANTIGONE dominates SCIP for model ($P_{2D-poly}$). The results also give credit for non-linear models, as paired with the adequate solver can lead to competitive CT. For instance, for the 1-HUC (**Table 5**), when solver BARON is used, configurations with model ($P_{5PL-bin}$) have a proportion of configurations solved and an avg-CT very similar to configurations relative to model (P^3_{pwl}) solved with CPLEX.

More details are given directly on the solvers behaviour in **Table 5** and **Table 6**. ANTIGONE seems to be less efficient than other solvers for the 1-HUC. This is especially notable for model ($P_{5PL-bin}$). BARON is on average the most efficient solver for models for which it has results. COUENNE is moderately effective and it seems like there is no model that is significantly difficult to solve, in comparison to other solvers. LINDOGlobal is pretty efficient for models (P_{op}) and ($P_{2D-poly}$). However, it is the least efficient for (P_{bilin}). For models ($P_{5PL-bin}$), LINDOGlobal is very efficient for the 1-HUC, but not for the fixed-head 1-HUC. The opposite can be noticed for model ($P_{5PL-max}$). SCIP is very efficient for models ($P_{5PL-bin}$), ($P_{5PL-max}$) and (P_{bilin}). However, it does not perform well with model ($P_{2D-poly}$).

	ANTIGONE		BARON		COUENNE		LINDOGlobal		SCIP		CPLEX	
Model	%S	avg-t	%S	avg-t	%S	avg-t	%S	avg-t	%S	avg-t	%S	avg-t
($P_{5PL-max}$)	NS		NS		NS		38.9	616.36	**86.1**	**606.87**	NS	
($P_{5PL-bin}$)	19.4	824.84	**88.9**	**25.63**	69.4	163.23	86.1	446.92	**88.9**	218.00	NS	
($P_{2D-poly}$)	44.4	**2.51**	**88.9**	31.32	63.9	91.27	80.6	30.36	19.4	828.72	NS	
(P_{op})	50.0	235.24	**83.3**	258.94	52.8	439.83	75.0	375.10	69.4	**44.23**	NS	
(P_{bilin})	75.0	0.10	**100.0**	**0.08**	97.2	0.25	75.0	4.72	**100.0**	0.17	NS	
(P^3_{pwl})	NR		NR		NR		NR		NR		**94.4**	**19.07**
(P^2_{pwl})	NR		NR		NR		NR		NR		**100.0**	**0.71**
(P^1_{pwl})	NR		NR		NR		NR		NR		**100.0**	**0.02**

Table 5: Proportion of configurations for each model solved by each solver and related average CT for the 1-HUC

Previous results are presented with respect to the *virtual best solver* (VBS). However, in a practical case it may not be convenient to use the VBS, as it could be difficult to have access to as many solvers. **Tables 7** and **8** show the proportion of instances where a solver is the VBS, for each model. For model ($P_{5PL-max}$), SCIP is the VBS in a large majority of configurations. For linear models (P^1_{pwl}), (P^2_{pwl}) and (P^3_{pwl}) for the 1-HUC and models (P_{bilin}), (P_{op}) for the fixed-head 1-HUC, CPLEX is the VBS as it is the only linear solver

Model	ANTIGONE		BARON		COUENNE		LINDOGlobal		SCIP		CPLEX	
	%S	avg-t	%S	avg-t	%S	avg-t	%S	avg-t	%S	avg-t	%S	avg-t
$(P_{5PL-max})$	NS		NS		NS		**97.2**	159.96	94.4	**0.13**	NS	
$(P_{5PL-bin})$	30.6	856.07	**100.0**	13.13	**100.0**	109.84	61.1	58.80	**100.0**	**2.95**	NS	
$(P_{2D-poly})$	72.2	**0.15**	**100.0**	0.51	100.0	0.87	**100.0**	1.36	**100.0**	100.31	NS	
(P_{op})	NR		NR		NR		NR		NR		**91.7**	**0.01**
(P_{bilin})	NR		NR		NR		NR		NR		**100.0**	**0.00**
(P^3_{pwl})	NR		NR		NR		NR		NR		**100.0**	**0.05**
(P^2_{pwl})	NR		NR		NR		NR		NR		**100.0**	**0.02**
(P^1_{pwl})	NR		NR		NR		NR		NR		**100.0**	**0.01**

Table 6: Proportion of configurations for each model solved by each solver and related average CT for the fixed-head 1-HUC

considered. For models $(P_{5PL-bin})$, $(P_{2D-poly})$, (P_{bilin}) and (P_{op}) for the 1-HUC and models $(P_{5PL-bin})$ and $(P_{2D-poly})$ for the fixed-head 1-HUC, BARON seems to be the VBS in most cases. We see two exceptions, for model (P_{bilin}) for the 1-HUC and model $(P_{2D-poly})$ for the fixed-head 1-HUC, ANTIGONE is more frequently the VBS. However, **Tables 5** and **6** show that in aforementioned cases, the problem is solved very quickly. Thus it is possible that ANTIGONE is quicker than BARON only on easy configurations, and may only be quicker to startup. Besides, ANTIGONE does not solve every configuration in both cited cases, while BARON does. To summarize, only one solver could be considered per model to have most of the time the VBS: CPLEX for linear models, SCIP for model $(P_{5PL-max})$ and BARON for all other cases.

Model	ANTIGONE	BARON	COUENNE	LINDOGlobal	SCIP	CPLEX
$(P_{5PL-max})$	NS	NS	NS	16.1	**83.9**	NS
$(P_{5PL-bin})$	0.0	**54.6**	30.3	15.2	0.0	NS
$(P_{2D-poly})$	12.5	**84.4**	3.1	0.0	0.0	NS
(P_{op})	0.0	**90.6**	0.0	6.3	3.1	NS
(P_{bilin})	**63.9**	33.3	0.0	0.0	2.8	NS
(P^3_{pwl})	NR	NR	NR	NR	NR	**100.0**
(P^2_{pwl})	NR	NR	NR	NR	NR	**100.0**
(P^1_{pwl})	NR	NR	NR	NR	NR	**100.0**

Table 7: Proportion of configurations for each model where a solver is the VBS for the 1-HUC

In **Appendix D** an analysis of the impact of each characteristic of a 1-HUC instance is described.

5.6 General modeling recommendations from numerical experiments

From the results, three types of models stand out: (P_{pwl}), (P_{op}) and $(P_{2D-poly})$ both for the 1-HUC and the fixed-head 1-HUC. We list them hereafter giving for each of them their main strengths and weaknesses. Firstly, (P_{pwl}) usually provides a good trade-off between CT and AE. However the proper number of

Model	ANTIGONE	BARON	COUENNE	LINDOGlobal	SCIP	CPLEX
$(P_{5PL-max})$	NS	NS	NS	5.6	**94.4**	NS
$(P_{5PL-bin})$	2.8	**72.2**	0.0	13.9	11.1	NS
$(P_{2D-poly})$	**52.8**	33.3	0.0	13.9	0.0	NS
(P_{op})	NR	NR	NR	NR	NR	**100.0**
(P_{bilin})	NR	NR	NR	NR	NR	**100.0**
(P^3_{pwl})	NR	NR	NR	NR	NR	**100.0**
(P^2_{pwl})	NR	NR	NR	NR	NR	**100.0**
(P^1_{pwl})	NR	NR	NR	NR	NR	**100.0**

Table 8: Proportion of configurations for each model where a solver is the VBS for the fixed-head 1-HUC

pieces cannot always be deduced in advance. Consequently, a trial and error procedure may be necessary to determine a piecewise linear function with a good trade-off. Secondly, model (P_{op}) can lead to the smallest AE, and it can be solved faster than the sophisticated models $(P_{5PL-bin})$ and $(P_{5PL-max})$. The drawback is that in the case of an instance with equality constraints there may not be a feasible solution for model (P_{op}). Thirdly, model $(P_{2D-poly})$ is similar to (P^2_{pwl}) in terms of AE and CT, meaning it is also a good trade-off. However, model $(P_{2D-poly})$ sometimes fails to find a solution, even if there is a feasible solution with the model. This illustrates the intrinsic difficulties of the current solvers for some non-linear models (see also the case of $(P_{HD-poly})$ described in **Section 5.4**).

The choice of the solver impacts the CT and the proportion of instances solved. The results indicate that BARON is the most efficient non-linear solver when the model is supported, otherwise SCIP is the most efficient one. For the three retained models, solver BARON is the most efficient for the non-linear ones (model (P_{op}) for the 1-HUC and $(P_{2D-poly})$ in any case), and a specialized MILP solver should be considered for the linear ones (model (P_{op}) for the fixed-head 1-HUC and (P_{pwl}) in any case).

6 Conclusion

In this paper various non-linear and linear modeling alternatives to solve a non-linear problem are compared, in terms of feasibility, approximation error, distance to the best recalculated value and computational time. The considered non-linear problem is the 1-HUC, featuring two non-linearities: a one-dimensional concave function, and a two-dimensional non-convex and non-concave function. A common special case of the 1-HUC, the fixed-head 1-HUC, is also considered, featuring a single non-linearity: a one-dimensional non-convex and non-concave function. A close to the physics non-linear model is defined for the 1-HUC and for the fixed-head 1-HUC. However, this model features too difficult non-linearities to be solved in a reasonable time, even for small instances. Seven alternative models are proposed, the focus being to represent the non-linearities of both the 1-HUC and the fixed-head 1-HUC. These mod-

els cover a large panel of modeling alternatives, including the common models for the 1-HUC from the literature, but also new models with uncommon non-linear functions. Several sets of instances with different characteristics of the 1-HUC and the fixed-head 1-HUC are solved with each of the proposed model, using five global solvers, and one linear solver. The results show that three of the seven models stand out as the most appealing, offering the best trade-off between computational time, approximation error and feasibility. As the computing time of a non-linear model highly depends on the available global solver, preferred solvers are also highlighted for these three models.

As future research, refining the three most efficient models revealed by the present study, via advanced or dedicated solution methods, is promising. First, the use of logarithmic disjunctive constraints [37] would benefit to all models, as it would reduce the number of variables. Second, model (P_{pwl}) can be improved by extending methods from the literature that optimize the number of breakpoints with an approximation guarantee in a PWL bounding framework [25]. Third, as model ($P_{2D-poly}$) features quadratic constraints, making use of quadratic programming techniques could improve the feasibility and the computational time. Finally, in opposition to the other presented models, the water flow is discretized in model (P_{op}) due to the finite set of operating points. Consequently there is a large combinatorics, which usually increases exponentially with the size of the instances. The use of combinatorial optimization methods, such as a polyhedral study could lead to smaller computing times.

References

[1] Ioannis P Androulakis. "MINLP: Branch and Bound Global Optimization Algorithm". In: *Encyclopedia of Optimization* (Jan. 2001), pp. 1415–1421.

[2] Pietro Belotti. *Couenne: a user's manual*. Tech. rep. Lehigh University, 2009.

[3] Pietro Belotti. "Bound reduction using pairs of linear inequalities". In: *Journal of Global Optimization* 56.3 (2013), pp. 787–819.

[4] Alberto Borghetti et al. "An MILP approach for short-term hydro scheduling and unit commitment with head-dependent reservoir". In: *IEEE Transactions on power systems* 23.3 (2008), pp. 1115–1124.

[5] Fani Boukouvala, Ruth Misener, and Christodoulos A Floudas. "Global optimization advances in mixed-integer nonlinear programming, MINLP, and constrained derivative-free optimization, CDFO". In: *European Journal of Operational Research* 252.3 (2016), pp. 701–727.

[6] João Paulo da Silva Catalão, Hugo Miguel Inácio Pousinho, and Victor Manuel Fernandes Mendes. "Mixed-integer nonlinear approach for the optimal scheduling of a head-dependent hydro chain". In: *Electric Power Systems Research* 80.8 (2010), pp. 935–942.

[7] IBM ILOG Cplex. "V12. 1: User's Manual for CPLEX". In: *International Business Machines Corporation* 46.53 (2009), p. 157.

[8] Keely L Croxton, Bernard Gendron, and Thomas L Magnanti. "A comparison of mixed-integer programming models for nonconvex piecewise linear cost minimization problems". In: *Management Science* 49.9 (2003), pp. 1268–1273.

[9] Joseph Czyzyk, Michael P Mesnier, and Jorge J Moré. "The NEOS server". In: *IEEE Computational Science and Engineering* 5.3 (1998), pp. 68–75.

[10] Claudia D'Ambrosio, Andrea Lodi, and Silvano Martello. "Piecewise linear approximation of functions of two variables in MILP models". In: *Operations Research Letters* 38.1 (2010), pp. 39–46.

[11] Jacques Desrosiers and Marco E Lübbecke. "Branch-price-and-cut algorithms". In: *Encyclopedia of Operations Research and Management Science. John Wiley & Sons, Chichester* (2011), pp. 109–131.

[12] E Edom et al. "On the impact of the power production function approximation on hydropower maintenance scheduling". In: *Les Cahiers du GERAD G–2020–22, GERAD* (2020).

[13] Thomas GW Epperly and Efstratios N Pistikopoulos. "A reduced space branch and bound algorithm for global optimization". In: *Journal of Global Optimization* 11.3 (1997), pp. 287–311.

[14] Javier García-González et al. "Under-relaxed iterative procedure for feasible short-term scheduling of a hydro chain". In: *2003 IEEE Bologna Power Tech Conference Proceedings,* vol. 2. 2003, 6 pp.

[15] Björn Geißler et al. “Using piecewise linear functions for solving MINLPs”. In: *Mixed integer nonlinear programming*. Vol. 154. The IMA Volumes in Mathematics and its Applications. Springer, 2012, pp. 287–314.

[16] Zvonimir Glasnovic and Jure Margeta. “The features of sustainable solar hydroelectric power plant”. In: *Renewable energy* 34.7 (2009), pp. 1742–1751.

[17] Paul G Gottschalk and John R Dunn. “The five-parameter logistic: a characterization and comparison with the four-parameter logistic”. In: *Analytical biochemistry* 343.1 (2005), pp. 54–65.

[18] Pascal Kerschke et al. “Leveraging TSP solver complementarity through machine learning”. In: *Evolutionary computation* 26.4 (2018), pp. 597–620.

[19] Ailsa H Land and Alison G Doig. “An automatic method for solving discrete programming problems”. In: *50 Years of Integer Programming 1958-2008*. Springer, 2010, pp. 105–132.

[20] Xiang Li et al. “Hydro unit commitment via mixed integer linear programming: A case study of the Three Gorges project, China”. In: *IEEE Transactions on Power Systems* 29.3 (2013), pp. 1232–1241.

[21] Ricardo M Lima et al. “On the computational studies of deterministic global optimization of head dependent short-term hydro scheduling”. In: *IEEE Transactions on Power Systems* 28.4 (2013), pp. 4336–4347.

[22] Youdong Lin and Linus Schrage. “The global solver in the LINDO API”. In: *Optimization Methods & Software* 24.4-5 (2009), pp. 657–668.

[23] SJPS Mariano et al. “Optimising power generation efficiency for head-sensitive cascaded reservoirs in a competitive electricity market”. In: *International Journal of Electrical Power & Energy Systems* 30.2 (2008), pp. 125–133.

[24] Ruth Misener and Christodoulos A Floudas. “ANTIGONE: algorithms for continuous/integer global optimization of nonlinear equations”. In: *Journal of Global Optimization* 59.2-3 (2014), pp. 503–526.

[25] Sandra Ulrich Ngueveu. “Piecewise linear bounding of univariate nonlinear functions and resulting mixed integer linear programming-based solution methods”. In: *European Journal of Operational Research* 275.3 (2019), pp. 1058–1071.

[26] SO Orero and MR Irving. “A genetic algorithm modelling framework and solution technique for short term optimal hydrothermal scheduling”. In: *IEEE Transactions on Power Systems* 13.2 (1998), pp. 501–518.

[27] Manfred Padberg and Giovanni Rinaldi. “A branch-and-cut algorithm for the resolution of large-scale symmetric traveling salesman problems”. In: *SIAM review* 33.1 (1991), pp. 60–100.

[28] Juan I Pérez and José R Wilhelmi. "Nonlinear self-scheduling of a single unit small hydro plant in the day-ahead electricity market". In: *Proc. of ICREPQ'07* (2007).

[29] Jesus A Rodriguez et al. "MILP formulations for generator maintenance scheduling in hydropower systems". In: *IEEE Transactions on Power Systems* 33.6 (2018), pp. 6171–6180.

[30] Hong S Ryoo and Nikolaos V Sahinidis. "A branch-and-reduce approach to global optimization". In: *Journal of global optimization* 8.2 (1996), pp. 107–138.

[31] Youcef Sahraoui, Pascale Bendotti, and Claudia d'Ambrosio. "Real-world hydro-power unit-commitment: Dealing with numerical errors and feasibility issues". In: *Energy* 184 (2019), pp. 91–104.

[32] Martin Savelsbergh. "A branch-and-price algorithm for the generalized assignment problem". In: *Operations research* 45.6 (1997), pp. 831–841.

[33] Sara Séguin, Pascal Côté, and Charles Audet. "Self-scheduling short-term unit commitment and loading problem". In: *IEEE Transactions on Power Systems* 31.1 (2015), pp. 133–142.

[34] Edward MB Smith and Constantinos C Pantelides. "A symbolic reformulation/spatial branch-and-bound algorithm for the global optimisation of nonconvex MINLPs". In: *Computers & Chemical Engineering* 23.4-5 (1999), pp. 457–478.

[35] M. Tawarmalani and N. V. Sahinidis. "A polyhedral branch-and-cut approach to global optimization". In: *Mathematical Programming* 103 (2 2005), pp. 225–249.

[36] Ragavan Vaidyanathan and Mahmoud El-Halwagi. "Global optimization of nonconvex MINLP's by interval analysis". In: *Global optimization in engineering design*. Springer, 1996, pp. 175–193.

[37] Juan Pablo Vielma and George L Nemhauser. "Modeling disjunctive constraints with a logarithmic number of binary variables and constraints". In: *Mathematical Programming* 128.1-2 (2011), pp. 49–72.

[38] Stefan Vigerske and Ambros Gleixner. "SCIP: Global optimization of mixed-integer nonlinear programs in a branch-and-cut framework". In: *Optimization Methods and Software* 33.3 (2018), pp. 563–593.

[39] Juan M Zamora and Ignacio E Grossmann. "A branch and contract algorithm for problems with concave univariate, bilinear and linear fractional terms". In: *Journal of Global Optimization* 14.3 (1999), pp. 217–249.

Appendices

A Solver description

ANTIGONE [24] is based on an sBB algorithms. The problem is reformulated in order to find special structures. Once the structures are found, the relaxation of the problem is solved. The search space is split and the process repeated until convergence of the upper and the lower bounds. Upper bounds are computed with local optimization algorithms. Only twice differentiable functions, that are not trigonometrical functions, are supported by ANTIGONE.

BARON [35] implements a deterministic Branch and Reduce algorithm. This algorithm contains constraint programming, interval analysis and duality techniques for tightening variables bounds. Heuristics, cutting planes and parallelism are combined with the Branch and Reduce algorithm. Trigonometrical functions and **max** functions are not supported.

COUENNE [2] implements an sBB with linearization, bound reductions and branching method. The main four components are: reformulation, separation of linearization cuts, branching rules and bound tightening methods. COUENNE only supports functions that can be reformulated into univariate functions and does not support function **max**.

LINDOGlobal [22] is the only solver that does not directly implements an sBB algorithm. Instead, it implements a branch and cut algorithm that breaks the model into sub-problems. The sub-problems are further split until each sub-problem is convex. The sub-problems are then solved with a BB or sBB algorithm. LINDOGlobal supports most non-linearities, and binary operators such as AND, OR and NOT.

SCIP [38] implements an sBB, where the non-linearities are represented within graphs. These graphs help finding convex non-linearities, and reformulating the non-linear functions. During the solving process, SCIP also adds various cuts, depending on the non-linearities. Bound tightening methods are also applied. Trigonometrical functions are not supported by SCIP and it is the only solver which requires a linear objective function.

CPLEX [7] implements a quite effective multipurpose Branch and Cut algorithm, which generates automatically various cuts [7]. Furthermore it is paired with pre-processing and heuristics.

B Five parameters logistic function

A $\mathbf{5PL}$ is the following function, where x is a variable and y_1 to y_5 the parameters:

$$5PL(x, y_1, y_2, y_3, y_4, y_5) = y_4 + \frac{-y_4}{(1 + (\frac{x - y_1}{y_3})^{y_2})^{y_5}}$$

In the context of the 1-HUC, variable x is the water-flow d_t. The $\mathbf{5PL}$ has a shape similar to a more common function, the sigmoid:

$$sig(x, y'_1, y'_2, y'_3) = \frac{y'_3}{1 + e^{-y'_2(x-y'_1)}}$$

The advantages of the $\mathbf{5PL}$ is that it is more flexible than a sigmoid. The sigmoid is necessarily symmetric with respect to its inflection point, whereas $\mathbf{5PL}$ is not. However, a $\mathbf{5PL}$ function is not defined if $x < y_1$, which can occur when representing a turbine by a $\mathbf{5PL}$ function. To adapt the $\mathbf{5PL}$ function to the use case of the 1-HUC, it is possible to insert a **max** function inside the $\mathbf{5PL}$ function as follows:

$$5PL(x, y_1, y_2, y_3, y_4, y_5) = y_4 + \frac{-y_4}{\left(1 + \left(\frac{\mathbf{max}(0, x-y_1)}{y_3}\right)^{y_2}\right)^{y_5}}$$

With this modification, if $x < y_1$ then the $\mathbf{5PL}$ function is equal to $y_4 + (-y_4/1) = 0$, if $x \geq y_1$, the $\mathbf{5PL}$ has the same behaviour as previously defined.

C Instances description

The instances are derived from the following parameter sets A and B, by changing the value of only one parameter at a time. The idea is to evaluate the impact of the parameters on the resolution and the solution with multiple metrics. **Table 9** shows the parameters of each parameter set.

Parameter set A	Parameter set B
$V_0^1 = 500$, $V_0^2 = 200$	$V_0^1 = 90$, $V_0^2 = 10$
$T = 4$	$T = 4$
$\overline{V}_t^1 = 1000$, $\underline{V}_t^1 = 0\ \forall t \leq T$	$\overline{V}_t^1 = 100$, $\underline{V}_t^1 = 0\ \forall t \leq T$
$\overline{V}_t^2 = 500$, $\underline{V}_t^2 = 0\ \forall t \leq T$	$\overline{V}_t^2 = 90$, $\underline{V}_t^2 = 0\ \forall t \leq T$
$\underline{D} = 0$, $\overline{D} = 25$	$\underline{D} = 0$, $\overline{D} = 8$
$\underline{P}_t = 0$, $\overline{P}_t = 15\ \forall t \leq T$	$\underline{P}_t = 0$, $\overline{P}_t = 32\ \forall t \leq T$
$\Phi^1 = 230$, $\Phi^2 = 0$	$\Phi^1 = 850$, $\Phi^2 = 0$
$\Lambda = [0.2, 0.15, 0.1, 0.2]$	$\Lambda = [0.1, 0.2, 0.5, 0.4]$
$A_t^1 = A_t^2 = 0\ \forall t \leq T$	$A_t^1 = A_t^2 = 0\ \forall t \leq T$
$\gamma = [0, 0.1, 5, 0.7]$	$\gamma = [100, 0.2, 2, 0.6]$

Table 9: Parameter sets

The modified characteristics are the following:

- Size of the instance
- Equality constraints

- Number of inflection points of the non-linear function
- Degree of non-linearity of the function
- Sensitivity of the decision variables to the non-linear effect

In these parameter sets, the maximum and minimum volumes are artificially large, to see the impact of each characteristic. Below, we justify the choice for each characteristic and explain how the changes are instantiated on the 1-HUC. Note that every instance is built such that there is at least one feasible solution with continuous water flows d_t.

C.1 Size of the instance

Larger instances are in general harder to solve as they contain more variables and inequalities. In the case of the 1-HUC, larger instances considered will have a larger number of time periods T. Increasing T exponentially increases the number of feasible solutions. Three instances are considered, A-T-1 to A-T-3 (resp. B-T-1 to B-T-3) corresponding to the variations of the parameter set A (resp. B) with 4, 7 and 10 time periods T. To take into account more time periods, prices are supplemented as follows: Λ = [0.2, 0.15, 0.1, 0.2, 0.1, 0.05, 0.1, 0.2, 0.15, 0.05] (resp. Λ = [0.1, 0.2, 0.5, 0.4, 0.3, 0.2, 0.3, 0.5, 0.4, 0.2]). These instances are such that the volume of each reservoir can not reach the maximum or minimum volume. The water flow will not be affected by the bounds on the volume, in contrary to some other sets of instances.

C.2 Equality constraints

Equality constraints can highly affect the resolution. Indeed, equality constraints drastically reduce the number of feasible solutions and can also be hard to satisfy. Moreover, depending on the approximation used in the model, equality constraints may lead to non efficient solutions. In the case of the 1-HUC, target volumes are equality constraints, when $\underline{V}_t^1 = \overline{V}_t^1$ for a time period t. Six instances are considered, A-E-1 to A-E-6 (resp. B-E-1 to B-E-6) which are variations of parameter set A (resp. B), where target volumes are only for the last time period T. For A-E-1 to A-E-3 (resp. B-E-1 to B-E-3) the target volumes are 480, 450 and 420 (resp. 80, 70 and 60). For A-E-4 to A-E-6 (resp B-E-4 to B-E-6), the target volumes are 500 (resp. 90), but the additional intake of water at the last time period are 20, 50 and 80 (resp. 10, 20, 30). One can notice that for instance A-E-1, the difference between the initial and the target volume is 20, while for instance A-E-4 it is 0, but the additional intake of water is 20. Thus, feasible solutions for A-E-1 are feasible solutions for A-E-4 and vice-versa. Instances A-E-2 and A-E-5, B-E-1 and B-E-4 and so on are built similarly.

C.3 Number of inflection points of the non-linear function

With a different number of inflection points, the shape of a non-linear function is changed, which can lead to more local optimal solutions, or less efficient under-estimators. The functions used to under-estimate and approximate the functions are also changed. Thus, the resolution and approximation error could be impacted by the number of inflection points of the non-linear function. In the case of the 1-HUC, the number of inflection points of the power function can be changed by defining a larger number of smaller turbines. We still have the same maximum power and maximum water flow, only the shape of the power function is different. Three instances are considered, A-N-1 to A-N-3 (resp. B-N-1 to B-N-3) which are variations of the parameter set A (resp. B) with 2, 4 and 6 turbines.

C.4 Degree of non-linearity of the non-linear function

As for the number of inflection points, changing the degree of non-linearity is another way to change the shape of a function. Thus, the resolution and approximation error could be affected by the degree of non-linearity of the functions. In the case of the 1-HUC, one way to increase the non-linearity of the power function is to change the water flow when each turbine starts and stops, increasing or reducing the degree of curvature for each concave part of the function, as represented by. The resulting power function for the unit can be quasi-linear or have a high degree of non-linearity. In addition, it also changes the domain of some variables. Six instances are considered, A-D-1 to A-D-6 (resp. B-D-1 to B-D-6) being variations of parameter set A (resp. B). Instances A-D-1 and A-D-4 feature a quasi linear function, with $\overline{D} = 22$ and $\overline{P}_t = 14.5$, $\forall t \leq T$, instances A-D-2 and A-D-5 correspond to a non-linear function, with $\overline{D} = 25$ and $\overline{P}_t = 15$, and instances A-D-3 and A-D-6 use a very non-linear function, with $\overline{D} = 28$ and $\overline{P}_t = 16$. The target volume for instances A-D-4 to A-D-6 is 460. Similarly, instances B-D-1 and B-D-4 feature a quasi linear function, with $\overline{D} = 6$ and $\overline{P}_t = 28$, instances B-D-2 and B-D-5 feature a non-linear function, with $\overline{D} = 8$ and $\overline{P}_t = 32$, and instances B-D-3 and B-D-6 feature a very non-linear function, with $\overline{D} = 10$ and $\overline{P}_t = 34$. The target volume for instances B-D-4 to B-D-6 is 75.

C.5 Sensitivity of the decision variables to the non-linear effect

Depending on the problem, decision variables can have a very large, or very small impact on the non-linearities. When the impact is small, it is possible that some simplifications of the problem would not induce large approximation errors. In the case of the 1-HUC, the sensitivity of the decision variables to the non-linear effect can change by considering larger or smaller reservoirs. Two instances are considered, A-S-1 and A-S-2 (resp. B-S-1 and B-S-2) are variations of parameter set A (resp. B). Instance A-S-2 is similar to A-S-1, but has all

initial, maximal and minimal volumes multiplied by 100, and supplemented prices $\Lambda = [0.005, 0.00375, 0.0025, 0.005]$. Analogously, B-S-2 is similar to B-S-1 with initial, maximal and minimal volumes all multiplied by 100, and adapted prices $\Lambda = [0.005, 0.01, 0.025, 0.02]$. The unit prices Λ are reduced in order to obtain similar solutions for instances A-S-1 and A-S-2 (resp. B-S-1 and B-S-2).

One can compute bounds on the variation of the volume, by calculating the maximum and minimum water processed while respecting the capacities. These bounds give an interval for the final volume in the reservoirs. It is then possible to compute the maximum difference in terms of volume between two feasible solutions, and compare it to the capacity of the reservoirs in order to predict if the instance might induce high volume variations or not. The sensitivity S can be computed as follows:

$$S = \frac{\overline{D} \times T - \underline{D} \times T}{\min(\overline{V}_T^1 - \underline{V}_T^1, \overline{V}_T^2 - \underline{V}_T^2)}$$

For instance A-S-1, the sensitivity is $100/500 = 0.2$, for instance A-S-2: 0.002, for instance B-S-1: 0.36 and for instance B-S-2: 0.0036. Note that the parameter set A (resp. B) has the same sensitivity as instance A-S-1 (resp. B-S-1).

Table 9 summarizes these instances and their characteristics.

Instances	Characteristics	Modified parameter
A-T-1 to A-T-3 B-T-1 to B-T-3	Size of the instance	Number of time periods
A-E-1 to A-E-6 B-E-1 to B-E-6	Equality constraints	Different target volumes, with and without additional intakes of water
A-N-1 to A-N-3 B-N-1 to B-N-3	Number of inflection points	Number of turbines
A-D-1 to A-D-6 B-D-1 to B-D-6	Degree of non-linearity	Quasi-linear, non-linear or very linear function, with and without target volumes
A-S-1, A-S-2, B-S-1, B-S-2	Sensitivity of the decision variables	Magnitude of the sensitivity

Table 10: Instance characteristics

D Analysis of the impact of the instance characteristics

Let us analyse the impact of each characteristic of a 1-HUC instance on the resolution. The tables related to the results described in the following section are in **Appendix E**

D.1 Size of the instance

Changing the number of time periods (instances A-T-1 to A-T-3 and B-T-1 to B-T-3) has a big effect on the resolution. Indeed, we see from **Table 11** and **Table 12** that configurations with more time periods require a drastically

increased CT compared to configurations with fewer time periods. The most salient case is for the 1-HUC where with $T = 10$ the only configurations solved under three hours by their VBS are with one of the following four models: $(P_{2D-poly})$, (P_{bilin}), (P^1_{pwl}) and (P^2_{pwl}). Moreover, with $T = 7$ configurations with model $(P_{5PL-max})$ are never solved by their VBS within three hours. The AE also increases for configurations with instances with more time periods. It is especially visible for the fixed-head 1-HUC, with $T = 10$ the minimal AE is around 20% using models $(P_{5PL-max})$, and the average AE are around 40% at least.

As there are more time periods, more variables and inequalities are introduced, exponentially increasing the number of feasible solutions. The reason why the AE increases is due to the fact that errors are propagated through the time periods. Also, for the fixed-head 1-HUC, more time periods mean, in general, more water processed. The volume varies with a higher magnitude from the initial volume when there are more time periods, leading to larger AE when considering a fixed head.

D.2 Equality constraints

Taking fixed target volumes (instances A-E-1 to A-E-6 , B-E-1 to B-E-6, A-D-1 to A-D-6 and B-D-1 to B-D-6) has a non-homogeneous impact on the resolution. From **Table 13** and **Table 14** we notice that configurations with target volumes reduce the CT required for the 1-HUC, compared to configurations without target volumes. For the AE, we notice multiple behaviours. For most models, configurations with target volumes yields to smaller average AE, but higher maximal AE, compared to configurations without target volumes. Non-represented results also showed that ANTIGONE solves less than 25% of configurations with target volumes whereas it solves more than 60% of configurations without target volumes. A similar but less marked behaviour is noticed for COUENNE.

The decreased CT is probably due to the fact that fewer solutions are feasible. The reduced AE are due to the target volume being very close to the initial volume for some instances. Less volume is processed, meaning a smaller power, and smaller AE. Besides, for the fixed-head 1-HUC, it also means less errors due to the fixed head, as the volume may not vary to much from the initial volume. We notice that some configurations with model (P_{op}) are not solved, for both the 1-HUC and the fixed-head 1-HUC. This is because the target volume may not be reachable with the finite set of water flows.

D.3 Degree of non-linearity

Changing the non-linearity of the power function (instances A-D-1 to A-D-6 and B-D-1 to B-D-6) can have an impact on the CT and the AE in the case of the 1-HUC, but only on the AE for the fixed-head 1-HUC. From **Table 15** and **Table 16**, we notice that configurations with pronounced non-linearities have larger CT for the 1-HUC than configurations with quasi-linear functions.

The configurations with non-linear models also have larger AE with pronounced non-linear functions, for both the 1-HUC and the fixed-head 1-HUC. The AE for configurations with linear model is not affected. We also see that all the configurations with model ($P_{2D-poly}$) are infeasible with every solver when the instance has a pronounced non-linear function, even if there exist feasible solutions for the instance.

The general increase of the AE for non-linear models can be explained by two reasons. Firstly, by instance construction, the turbines are the same for every instances, and the water-flow interval for each turbine is changed in order to have a different degree of non-linearity. As such, it is possible that lesser non-linear functions can better approximate the function of the original model on a larger interval. Secondly, a highly non-linear function can be harder to approximate by simpler functions, leading to larger AE for every model.

D.4 Number of inflection points

Changing the number of turbines (instances A-N-1 to A-N-3 and B-N-1 to B-N-3) has only a noticeable impact on models representing each turbine explicitly, namely ($P_{5PL-max}$), ($P_{5PL-bin}$) and ($P_{2D-poly}$). Indeed, **Table 17** and **Table 18** show the increased CT required for configurations with these models and with instances with more turbines. Also, increased number of turbines reduces the degree of non-linearity. Thus it is possible to see similar behaviours as when changing the degree of non-linearity.

The reason why the CT increases for configurations with one of the four mentioned models and an instance with many turbines is because as they represent each turbine explicitly, more variables and inequalities are required.

D.5 Sensitivity of the decision variables to the non-linear effect

In order to have negligible variation of the volume, the volumes can be set to larger values than the water flows. **Table 19** and **Table 20** show that the CT tends to be smaller for configurations with large volumes compared to configurations with smaller volumes. Larger volumes usually lead to an improvement of the AE for the fixed-head 1-HUC. However, the maximal AE of PWL models can be very large with large volumes. More precisely, with a PWL models, half the configurations has a large AE, and the other half has a smaller AE, compared to configurations with small volumes

The improvement of AE for the fixed-head 1-HUC is because with small variations, the volume is very similar to the initial volume at any time period. The AE from the fixed-head becomes very small. The high AE of the PWL models can be explained as follows. These models only consider a family of univariate PWL function for a finite set of possible volumes. It is then possible that the volume is never similar to the volumes used by this family of functions.

E Numerical experiments when partitioning instances

- %S: proportion of configuration solved
- min-CT, max-CT, avg-CT: minimum, maximum, and average CR for every solved configurations.
- min-AE, max-AE, avg-AE: minimum, maximum and average AE for every solved configurations. (model, solver)

E.1 Size of the instance

Table 11 and **Table 12** represents the proportion of configurations with instances with 4, 7 and 10 time periods and each model solved by their VBS, and related minimum, maximum and average CT and AE.

Instances	Model	%S	min-t	max-t	avg-t	min-e	max-e	avg-e
T=4	$(P_{5PL-max})$	100.0	103.34	186.38	144.86	0.2	0.4	0.3
	$(P_{5PL-bin})$	100.0	15.28	28.47	21.88	0.2	0.4	0.3
	$(P_{2D-poly})$	100.0	0.43	0.69	0.56	0.8	3.9	2.4
	(P_{op})	100.0	1.47	2.25	1.86	0.3	0.4	0.3
	(P_{bilin})	100.0	0.03	0.05	0.04	24.3	26.1	25.2
	(P^3_{pwl})	100.0	0.4	7.06	3.73	1.3	3.5	2.4
	(P^2_{pwl})	100.0	0.09	0.18	0.14	5.8	6.5	6.2
	(P^1_{pwl})	100.0	0.01	0.02	0.01	11.8	67.9	39.9
T=7	$(P_{5PL-max})$	0	-	-	-	-	-	-
	$(P_{5PL-bin})$	100.0	10 367.97	10 367.97	10 367.97	0.4	0.4	0.4
	$(P_{2D-poly})$	100.0	8.31	36.18	22.25	0.8	3.3	2.0
	(P_{op})	100.0	78.69	574.13	326.41	4.0	14.3	9.2
	(P_{bilin})	100.0	0.06	0.26	0.16	24.1	31.3	27.7
	(P^3_{pwl})	100.0	39.12	39.12	39.12	1.3	1.3	1.3
	(P^2_{pwl})	100.0	0.15	2.88	1.51	5.7	8.7	7.2
	(P^1_{pwl})	100.0	0.03	0.06	0.04	27.9	67.9	47.9
T=10	$(P_{5PL-max})$	0	-	-	-	-	-	-
	$(P_{5PL-bin})$	0	-	-	-	-	-	-
	$(P_{2D-poly})$	100.0	44.71	344.08	194.39	0.8	14.1	7.5
	(P_{op})	100.0	7062.75	7062.75	7062.75	78.6	78.6	78.6
	(P_{bilin})	100.0	0.07	0.09	0.08	40.4	46.3	43.3
	(P^3_{pwl})	100.0	558.73	558.73	558.73	1.1	1.1	1.1
	(P^2_{pwl})	100.0	0.46	19.05	9.76	7.4	14.0	10.7
	(P^1_{pwl})	100.0	0.05	0.16	0.11	32.4	66.8	49.6

Table 11: Proportion of configurations solved with their VBS, CT and AE statistics for the 1-HUC for different number of time periods (instances A-T-1 to A-T-3 and B-T-1 to B-T-3)

Instances	Model	%S	min-t	max-t	avg-t	min-e	max-e	avg-e
T=4	$(P_{5PL-max})$	100.0	0.11	0.12	0.11	10.7	20.6	15.7
	$(P_{5PL-bin})$	100.0	0.25	0.4	0.33	10.7	20.6	15.7
	$(P_{2D-poly})$	100.0	0.06	0.08	0.07	14.6	27.3	20.9
	(P_{op})	100.0	0.0	0.01	0.01	10.3	21.0	15.7
	(P_{bilin})	100.0	0.0	0	0.0	41.0	66.6	53.8
	(P^3_{pwl})	100.0	0.02	0.02	0.02	20.2	21.9	21.0
	(P^2_{pwl})	100.0	0.01	0.01	0.01	19.9	21.8	20.9
	(P^1_{pwl})	100.0	0.01	0.01	0.01	21.9	22.2	22.0
T=7	$(P_{5PL-max})$	100.0	0.11	0.14	0.12	12.4	32.1	22.2
	$(P_{5PL-bin})$	100.0	2.77	10.14	6.46	18.6	32.1	25.4
	$(P_{2D-poly})$	100.0	0.06	0.1	0.08	23.6	43.4	33.5
	(P_{op})	100.0	0.01	0.01	0.01	17.8	33.1	25.5
	(P_{bilin})	100.0	0.0	0	0.0	60.2	115.6	87.9
	(P^3_{pwl})	100.0	0.03	0.03	0.03	31.5	32.1	31.8
	(P^2_{pwl})	100.0	0.01	0.01	0.01	30.8	31.6	31.2
	(P^1_{pwl})	100.0	0.01	0.01	0.01	30.2	35.7	33.0
T=10	$(P_{5PL-max})$	100.0	0.11	0.15	0.13	19.6	59.2	39.4
	$(P_{5PL-bin})$	100.0	6.25	50.0	28.12	29.7	59.2	44.5
	$(P_{2D-poly})$	100.0	0.08	0.08	0.08	34.7	67.3	51.0
	(P_{op})	100.0	0.01	0.01	0.01	28.4	49.0	38.7
	(P_{bilin})	100.0	0.0	0	0.0	78.7	149.9	114.3
	(P^3_{pwl})	100.0	0.04	0.04	0.04	44.7	47.0	45.9
	(P^2_{pwl})	100.0	0.01	0.02	0.01	44.2	46.1	45.2
	(P^1_{pwl})	100.0	0.01	0.01	0.01	42.9	52.8	47.8

Table 12: Proportion of configurations solved with their VBS, CT and AE statistics for the fixed-head 1-HUC for different number of time periods (instances A-T-1 to A-T-3 and B-T-1 to B-T-3)

E.2 Equality constraints

Table 13 and **Table 14** represents the proportion of configurations with instances with and without target volumes and each model solved by their VBS, and related minimum, maximum and average CT and AE.

E.3 Degree of non-linearity

Table 15 and **Table 16** represents the proportion of configurations with instances with a quasi linear, a non-linear and a very non-linear function and each model solved by their VBS, and related minimum, maximum and average CT and AE.

E.4 Number of inflection points

Table 17 and **Table 18** represents the proportion of configurations with instances with 2, 6 and 6 turbines and each model solved by their VBS, and

Instances	Model	%S	min-t	max-t	avg-t	min-e	max-e	avg-e
No target volume	$(P_{5PL-max})$	100.0	51.99	212.62	148.12	0.0	12.3	1.9
	$(P_{5PL-bin})$	100.0	4.08	31.84	21.97	0.0	12.3	1.9
	$(P_{2D-poly})$	100.0	0.4	0.81	0.53	0.0	7.2	2.8
	(P_{op})	100.0	1.47	2.95	1.94	0.1	12.7	1.9
	(P_{bilin})	100.0	0.03	0.06	0.04	19.4	32.3	26.2
	(P^3_{pwl})	100.0	0.4	8.98	3.6	0.9	4.2	2.5
	(P^2_{pwl})	100.0	0.09	0.18	0.13	4.4	6.5	5.8
	(P^1_{pwl})	100.0	0.01	0.03	0.02	10.0	70.6	39.8
target volume	$(P_{5PL-max})$	100.0	0.66	399.96	94.73	0.0	9.3	1.1
	$(P_{5PL-bin})$	100.0	0.1	18.31	7.66	0.0	9.3	0.8
	$(P_{2D-poly})$	100.0	0.09	0.48	0.29	0.2	4.2	1.4
	(P_{op})	100.0	0.56	23.96	3.81	0.0	10.5	1.2
	(P_{bilin})	100.0	0.03	0.13	0.07	0.5	27.4	12.4
	(P^3_{pwl})	100.0	0.14	1.0	0.44	0.8	22.9	3.4
	(P^2_{pwl})	100.0	0.01	0.17	0.07	1.3	30.8	6.6
	(P^1_{pwl})	100.0	0.01	0.02	0.01	4.9	69.4	33.0

Table 13: Proportion of configurations solved with their VBS, CT and AE statistics for the 1-HUC with and without target volumes (instances A-T-1, B-T-1, A-E-1 to A-E-6, B-E-1 to B-E-6, A-D-1 to A-D-6 and B-D-1 to B-D-6)

Instances	Model	%S	min-t	max-t	avg-t	min-e	max-e	avg-e
No target volume	$(P_{5PL-max})$	100.0	0.1	0.12	0.11	7.3	24.1	15.9
	$(P_{5PL-bin})$	100.0	0.2	0.48	0.35	8.8	28.7	17.8
	$(P_{2D-poly})$	100.0	0.06	0.14	0.08	9.6	28.4	20.8
	(P_{op})	100.0	0.0	0.01	0.01	8.0	27.8	17.4
	(P_{bilin})	100.0	0.0	0	0.0	29.0	87.5	56.5
	(P^3_{pwl})	100.0	0.02	0.03	0.03	17.7	27.5	21.4
	(P^2_{pwl})	100.0	0.01	0.02	0.01	17.2	27.4	21.1
	(P^1_{pwl})	100.0	0.01	0.01	0.01	18.4	29.2	22.2
target volume	$(P_{5PL-max})$	100.0	0.1	0.93	0.2	2.0	19.5	7.6
	$(P_{5PL-bin})$	100.0	0.03	1.06	0.52	1.0	80.0	14.0
	$(P_{2D-poly})$	100.0	0.06	0.23	0.13	0.6	28.8	10.4
	(P_{op})	100.0	0.0	0.02	0.01	1.0	19.9	7.1
	(P_{bilin})	100.0	0.0	0	0.0	2.7	49.8	21.2
	(P^3_{pwl})	100.0	0.02	0.17	0.07	0.8	33.5	10.6
	(P^2_{pwl})	100.0	0.01	0.04	0.02	0.8	33.5	10.6
	(P^1_{pwl})	100.0	0.0	0.01	0.01	0.9	33.6	10.8

Table 14: Proportion of configurations solved with their VBS, CT and AE statistics for the fixed-head 1-HUC with and without target volumes (instances A-T-1, B-T-1, A-E-1 to A-E-6, B-E-1 to B-E-6, A-D-1 to A-D-6 and B-D-1 to B-D-6)

related minimum, maximum and average CT and AE.

Instances	Model	%S	min-t	max-t	avg-t	min-e	max-e	avg-e
Quasi linear	$(P_{5PL-max})$	100.0	2.41	399.96	162.15	0.0	0.5	0.3
	$(P_{5PL-bin})$	100.0	3.18	31.84	13.63	0.0	0.5	0.3
	$(P_{2D-poly})$	100.0	0.23	0.4	0.35	0.0	7.2	2.9
	(P_{op})	100.0	1.43	1.68	1.54	0.1	0.5	0.2
	(P_{bilin})	100.0	0.03	0.08	0.05	10.2	27.7	18.1
	(P^3_{pwl})	100.0	0.2	3.39	1.15	1.0	3.6	2.2
	(P^2_{pwl})	100.0	0.1	0.14	0.12	4.4	6.2	5.4
	(P^1_{pwl})	100.0	0.01	0.02	0.02	10.0	68.7	38.8
Non-linear	$(P_{5PL-max})$	100.0	43.21	239.32	148.84	0.2	0.4	0.3
	$(P_{5PL-bin})$	100.0	8.83	29.02	17.6	0.2	0.4	0.3
	$(P_{2D-poly})$	100.0	0.21	0.81	0.46	0.8	3.9	1.9
	(P_{op})	100.0	1.08	2.93	1.98	0.3	0.4	0.3
	(P_{bilin})	100.0	0.04	0.08	0.05	0.9	26.1	16.7
	(P^3_{pwl})	100.0	0.15	6.73	2.01	1.3	3.6	2.6
	(P^2_{pwl})	100.0	0.03	0.16	0.1	5.8	7.1	6.4
	(P^1_{pwl})	100.0	0.01	0.02	0.02	11.8	68.4	40.0
Very non-linear	$(P_{5PL-max})$	100.0	62.31	208.98	151.88	1.0	12.3	6.2
	$(P_{5PL-bin})$	100.0	0.56	27.81	15.87	0.0	12.3	5.7
	$(P_{2D-poly})$	0	-	-	-	-	-	-
	(P_{op})	100.0	0.83	3.58	2.29	0.9	12.7	6.3
	(P_{bilin})	100.0	0.03	0.1	0.06	7.8	32.3	21.7
	(P^3_{pwl})	100.0	0.37	8.98	2.87	0.9	4.2	2.6
	(P^2_{pwl})	100.0	0.09	0.17	0.14	5.2	7.2	6.3
	(P^1_{pwl})	100.0	0.02	0.03	0.02	10.0	70.6	40.0

Table 15: Proportion of configurations solved with their VBS, CT and AE statistics for the 1-HUC for different degree of non-linearity (instances A-D-1 to A-D-6 and B-D-1 to B-D-6)

E.5 Sensitivity of the decision variables to the non-linear effect

Table 19 and **Table 20** represents the proportion of configurations with instances with small and large volumes and each model solved by their VBS, and related minimum, maximum and average CT and AE.

Instances	Model	%S	min-t	max-t	avg-t	min-e	max-e	avg-e
Quasi linear	$(P_{5PL-max})$	100.0	0.1	0.13	0.11	7.1	18.4	10.2
	$(P_{5PL-bin})$	100.0	0.2	0.82	0.46	7.1	18.4	10.5
	$(P_{2D-poly})$	100.0	0.06	0.14	0.1	6.8	28.4	14.3
	(P_{op})	100.0	0.01	0.01	0.01	7.9	16.8	10.9
	(P_{bilin})	100.0	0.0	0	0.0	21.1	53.2	31.4
	(P^3_{pwl})	100.0	0.02	0.03	0.03	7.5	18.5	14.8
	(P^2_{pwl})	100.0	0.01	0.02	0.01	7.4	18.4	14.7
	(P^1_{pwl})	100.0	0.01	0.01	0.01	6.8	19.8	15.2
Non-linear	$(P_{5PL-max})$	100.0	0.11	0.17	0.12	6.2	20.6	11.7
	$(P_{5PL-bin})$	100.0	0.32	1.03	0.56	6.2	20.6	11.5
	$(P_{2D-poly})$	100.0	0.06	0.23	0.15	7.5	27.3	14.1
	(P_{op})	100.0	0.0	0.01	0.01	6.1	21.0	11.5
	(P_{bilin})	100.0	0.0	0	0.0	18.7	66.6	37.3
	(P^3_{pwl})	100.0	0.02	0.03	0.03	7.8	21.9	16.1
	(P^2_{pwl})	100.0	0.01	0.01	0.01	7.6	21.8	16.0
	(P^1_{pwl})	100.0	0.01	0.01	0.01	6.1	22.2	15.9
Very non-linear	$(P_{5PL-max})$	100.0	0.1	0.12	0.11	10.1	24.1	15.9
	$(P_{5PL-bin})$	100.0	0.35	1.01	0.56	10.3	28.7	19.9
	$(P_{2D-poly})$	100.0	0.06	0.19	0.11	6.8	26.8	15.7
	(P_{op})	100.0	0.01	0.02	0.01	5.0	27.8	18.7
	(P_{bilin})	100.0	0.0	0	0.0	26.1	87.5	52.6
	(P^3_{pwl})	100.0	0.03	0.12	0.05	8.1	27.5	18.8
	(P^2_{pwl})	100.0	0.01	0.03	0.02	8.0	27.4	18.5
	(P^1_{pwl})	100.0	0.01	0.01	0.01	12.2	29.2	20.2

Table 16: Proportion of configurations solved with their VBS, CT and AE statistics for the fixed-head 1-HUC for different degree of non-linearity (instances A-D-1 to A-D-6 and B-D-1 to B-D-6)

Instances	Model	%S	min-t	max-t	avg-t	min-e	max-e	avg-e
N=2	$(P_{5PL-max})$	100.0	103.34	186.38	144.86	0.2	0.4	0.3
	$(P_{5PL-bin})$	100.0	15.28	28.47	21.88	0.2	0.4	0.3
	$(P_{2D-poly})$	100.0	0.43	0.69	0.56	0.8	3.9	2.4
	(P_{op})	100.0	1.47	2.25	1.86	0.3	0.4	0.3
	(P_{bilin})	100.0	0.03	0.05	0.04	24.3	26.1	25.2
	(P^3_{pwl})	100.0	0.4	7.06	3.73	1.3	3.5	2.4
	(P^2_{pwl})	100.0	0.09	0.18	0.14	5.8	6.5	6.2
	(P^1_{pwl})	100.0	0.01	0.02	0.01	11.8	67.9	39.9
N=4	$(P_{5PL-max})$	100.0	1764.35	5988.44	3876.39	0.1	0.4	0.2
	$(P_{5PL-bin})$	100.0	13.75	63.73	38.74	0.1	0.4	0.2
	$(P_{2D-poly})$	100.0	2.76	3.47	3.12	0.4	3.1	1.8
	(P_{op})	100.0	0.49	1.57	1.03	0.0	0.3	0.1
	(P_{bilin})	100.0	0.04	0.05	0.04	19.6	22.1	20.9
	(P^3_{pwl})	100.0	0.75	5.73	3.24	0.3	3.3	1.8
	(P^2_{pwl})	100.0	0.11	0.15	0.13	4.0	6.4	5.2
	(P^1_{pwl})	100.0	0.02	0.02	0.02	12.0	70.9	41.5
N=6	$(P_{5PL-max})$	100.0	1055.49	1055.49	1055.49	0.0	0	0.0
	$(P_{5PL-bin})$	100.0	7.63	36.01	21.82	0.0	0.3	0.1
	$(P_{2D-poly})$	100.0	1.11	3.81	2.46	0.7	8.7	4.7
	(P_{op})	100.0	0.38	1.3	0.84	0.0	0.9	0.5
	(P_{bilin})	100.0	0.04	0.04	0.04	13.6	20.4	17.0
	(P^3_{pwl})	100.0	0.5	6.63	3.56	0.4	4.2	2.3
	(P^2_{pwl})	100.0	0.14	0.22	0.18	3.7	6.4	5.1
	(P^1_{pwl})	100.0	0.02	0.02	0.02	10.6	75.8	43.2

Table 17: Proportion of configurations solved with their VBS, CT and AE statistics for the 1-HUC for different number of turbines (instances A-N-1 to A-N-3 and B-N-1 to B-N-3)

Instances	Model	%S	min-t	max-t	avg-t	min-e	max-e	avg-e
N=2	$(P_{5PL-max})$	100.0	0.11	0.12	0.11	10.7	20.6	15.7
	$(P_{5PL-bin})$	100.0	0.25	0.4	0.33	10.7	20.6	15.7
	$(P_{2D-poly})$	100.0	0.06	0.08	0.07	14.6	27.3	20.9
	(P_{op})	100.0	0.0	0.01	0.01	10.3	21.0	15.7
	(P_{bilin})	100.0	0.0	0	0.0	41.0	66.6	53.8
	(P^3_{pwl})	100.0	0.02	0.02	0.02	20.2	21.9	21.0
	(P^2_{pwl})	100.0	0.01	0.01	0.01	19.9	21.8	20.9
	(P^1_{pwl})	100.0	0.01	0.01	0.01	21.9	22.2	22.0
N=4	$(P_{5PL-max})$	100.0	0.11	0.33	0.22	12.2	24.9	18.5
	$(P_{5PL-bin})$	100.0	0.14	0.76	0.45	14.4	22.4	18.4
	$(P_{2D-poly})$	100.0	0.06	0.09	0.07	10.3	11.2	10.8
	(P_{op})	100.0	0.01	0.01	0.01	11.6	22.0	16.8
	(P_{bilin})	100.0	0.0	0	0.0	36.7	56.7	46.7
	(P^3_{pwl})	100.0	0.02	0.03	0.03	22.7	23.4	23.0
	(P^2_{pwl})	100.0	0.01	0.01	0.01	22.8	23.0	22.9
	(P^1_{pwl})	100.0	0.01	0.01	0.01	21.7	26.5	24.1
N=6	$(P_{5PL-max})$	100.0	0.1	0.33	0.22	13.0	25.4	19.2
	$(P_{5PL-bin})$	100.0	0.03	0.82	0.42	24.7	90.2	57.5
	$(P_{2D-poly})$	100.0	0.06	0.07	0.07	13.9	15.6	14.8
	(P_{op})	100.0	0.0	0	0.0	10.3	24.0	17.1
	(P_{bilin})	100.0	0.0	0	0.0	35.0	48.4	41.7
	(P^3_{pwl})	100.0	0.02	0.03	0.03	23.1	25.1	24.1
	(P^2_{pwl})	100.0	0.01	0.01	0.01	23.5	24.9	24.2
	(P^1_{pwl})	100.0	0.01	0.01	0.01	22.2	23.6	22.9

Table 18: Proportion of configurations solved with their VBS, CT and AE statistics for the fixed-head 1-HUC for different number of turbines (instances A-N-1 to A-N-3 and B-N-1 to B-N-3)

Instances	Model	%S	min-t	max-t	avg-t	min-e	max-e	avg-e
Large variations	$(P_{5PL-max})$	100.0	103.34	186.38	144.86	0.2	0.4	0.3
	$(P_{5PL-bin})$	100.0	15.28	28.47	21.88	0.2	0.4	0.3
	$(P_{2D-poly})$	100.0	0.43	0.69	0.56	0.8	3.9	2.4
	(P_{op})	100.0	1.47	2.25	1.86	0.3	0.4	0.3
	(P_{bilin})	100.0	0.03	0.05	0.04	24.3	26.1	25.2
	(P^3_{pwl})	100.0	0.4	7.06	3.73	1.3	3.5	2.4
	(P^2_{pwl})	100.0	0.09	0.18	0.14	5.8	6.5	6.2
	(P^1_{pwl})	100.0	0.01	0.02	0.01	11.8	67.9	39.9
Small variations	$(P_{5PL-max})$	100.0	123.58	131.2	127.39	0.5	0.8	0.7
	$(P_{5PL-bin})$	100.0	7.2	16.72	11.96	0.5	0.8	0.7
	$(P_{2D-poly})$	100.0	0.26	0.36	0.31	3.7	17.5	10.6
	(P_{op})	100.0	0.18	0.43	0.3	0.5	1.0	0.8
	(P_{bilin})	100.0	0.05	0.06	0.06	9.6	29.8	19.7
	(P^3_{pwl})	100.0	0.05	0.11	0.08	2.7	171.9	87.3
	(P^2_{pwl})	100.0	0.01	0.01	0.01	10.8	173.6	92.2
	(P^1_{pwl})	100.0	0.01	0.02	0.01	86.3	100.9	93.6

Table 19: Proportion of configurations solved with their VBS, CT and AE statistics for the 1-HUC for small and large volumes (instances A-S-1, A-S-2, B-S-1 and B-S-2)

Instances	Model	%S	min-t	max-t	avg-t	min-e	max-e	avg-e
Large variations	$(P_{5PL-max})$	100.0	0.11	0.12	0.11	10.7	20.6	15.7
	$(P_{5PL-bin})$	100.0	0.25	0.4	0.33	10.7	20.6	15.7
	$(P_{2D-poly})$	100.0	0.06	0.08	0.07	14.6	27.3	20.9
	(P_{op})	100.0	0.0	0.01	0.01	10.3	21.0	15.7
	(P_{bilin})	100.0	0.0	0	0.0	41.0	66.6	53.8
	(P^3_{pwl})	100.0	0.02	0.02	0.02	20.2	21.9	21.0
	(P^2_{pwl})	100.0	0.01	0.01	0.01	19.9	21.8	20.9
	(P^1_{pwl})	100.0	0.01	0.01	0.01	21.9	22.2	22.0
Small variations	$(P_{5PL-max})$	100.0	0.14	0.18	0.16	0.1	1.2	0.7
	$(P_{5PL-bin})$	100.0	0.72	0.84	0.78	0.2	1.0	0.6
	$(P_{2D-poly})$	100.0	0.09	0.09	0.09	4.2	16.7	10.4
	(P_{op})	100.0	0.0	0	0.0	0.2	1.1	0.7
	(P_{bilin})	100.0	0.0	0	0.0	9.8	30.3	20.1
	(P^3_{pwl})	100.0	0.03	0.03	0.03	0.8	176.3	88.6
	(P^2_{pwl})	100.0	0.01	0.01	0.01	0.8	175.8	88.3
	(P^1_{pwl})	100.0	0.01	0.01	0.01	0.7	177.8	89.2

Table 20: Proportion of configurations solved with their VBS, CT and AE statistics for the fixed-head 1-HUC for small and large volumes (instances A-S-1, A-S-2, B-S-1 and B-S-2)