fermentation

Article

Model Identification of *E. coli* Cultivation Process Applying Hybrid Crow Search Algorithm

Olympia Roeva [1,2,*,†] and **Dafina Zoteva** [3,†]

[1] Department of Bioinformatics and Mathematical Modelling, Institute of Biophysics and Biomedical Engineering, Bulgarian Academy of Sciences, Acad. G. Bonchev Str., bl. 105, 1113 Sofia, Bulgaria
[2] Department of Mechatronic Bio/Technological Systems, Institute of Robotics, Bulgarian Academy of Science, Acad. G. Bonchev Str., bl. 2, 1113 Sofia, Bulgaria
[3] Department of Computer Informatics, Faculty of Mathematics and Informatics, Sofia University "St. Kliment Ohridski", 1164 Sofia, Bulgaria; dafinaz@fmi.uni-sofia.bg
* Correspondence: olympia@biomed.bas.bg
† These authors contributed equally to this work.

Abstract: Cultivation process (CP) modeling and optimization are ambitious tasks due to the nonlinear nature of the models and interdependent parameters. The identification procedures for such models are challenging. Metaheuristic algorithms exhibit promising performance for such complex problems since a near-optimal solution can be found in an acceptable time. The present research explores a new hybrid metaheuristic algorithm built upon the good exploration of the genetic algorithm (GA) and the exploitation of the crow search algorithm (CSA). The efficiency of the proposed GA-CSA hybrid is studied with the model parameter identification procedure of the *E. coli BL21(DE3)pPhyt109* fed-batch cultivation process. The results are compared with those of the pure GA and pure CSA applied to the same problem. A comparison with two deterministic algorithms, i.e., sequential quadratic programming (SQP) and the Quasi-Newton (Q-N) method, is also provided. A more accurate model is obtained by the GA-CSA hybrid with fewer computational resources. Although SQP and Q-N find a solution for a smaller number of function evaluations, the resulting models are not as accurate as the models generated by the three metaheuristic algorithms. The InterCriteria analysis, a mathematical approach to revealing certain relations between given criteria, and a series of statistical tests are employed to prove that there is a statistically significant difference between the results of the three stochastic algorithms. The obtained mathematical models are then successfully verified with a different set of experimental data, in which, again, the closest one is the GA-CSA model. The GA-CSA hybrid proposed in this paper is proven to be successful in the collaborative hybridization of GA and CSA with outstanding performance.

Keywords: model identification; *E. coli BL21(DE3)pPhyt109*; crow search algorithm; genetic algorithm; sequential quadratic programming; Quasi-Newton method

check for updates

Citation: Roeva, O.; Zoteva, D. Model Identification of *E. coli* Cultivation Process Applying Hybrid Crow Search Algorithm. *Fermentation* **2024**, *10*, 12. https://doi.org/10.3390/fermentation10010012

Academic Editor: Liang Yu

Received: 4 December 2023
Revised: 19 December 2023
Accepted: 21 December 2023
Published: 22 December 2023

1. Introduction

The conventional methods of optimization (parameter identification), such as the steepest descent, sequential quadratic programming (SQP) and Newton methods (Quasi-Newton (Q-N)), rely on the restrictions of the initial solution. These methods perform well for simple problems. Challenges emerge when dealing with complex tasks characterized by nonlinearity, high dimensionality, multimodality, prohibited regions induced by constraints, etc. [1]. Solutions to highly nonlinear problems usually require sophisticated optimization algorithms, as conventional algorithms may struggle to deal with such problems [2]. A current trend is to use nature-inspired algorithms due to their flexibility and effectiveness.

Metaheuristic algorithms have shown promising performance on such complicated tasks [3]. Both single and population-based metaheuristics stand out as effective alternatives to traditional optimization methods [4,5]. They tackle various optimization problems successfully [6,7]. Several published studies have compared optimization methods from different categories (classical/traditional vs. metaheuristic) to determine their effectiveness in solving different problems. The authors in [8] compared the traditional and metaheuristic methods when optimizing robot motion planning. The implementation of metaheuristic methods resulted in superior performance compared to the traditional methods. In [9], it was discussed that classical optimization methods such as Newton's approach, linear programming and quadratic programming, for problems with high nonlinearity and nonconvexity issues, cannot reach a global solution but only obtain local solutions.

Cultivation process (CP) modeling and optimization are demanding and ambitious tasks due to the nonlinear nature of the models and their mutually dependent parameters [10]. The challenging identification procedures for such models are usually handled by metaheuristic techniques, relying on the fact that they lead to a near-optimal solution in a tolerable amount of time [11].

Several metaheuristic algorithms, such as the genetic algorithm (GA) [12], cuckoo search algorithm (CS) [13] and ant colony optimization [14], have been effectively employed to identify parameters in CP models. Meanwhile, others, like the crow search algorithm (CSA) [15], have not yet been explored for this specific problem.

Multiple GA and CS [16] and GA-CS hybrid [17] methods have been proposed for CP modeling. The CS algorithm has been also used in [18] to model the process of biogas production. The sparrow search algorithm, whale optimization algorithm and African vulture optimization algorithm have been applied in [19] for the modeling of kombucha fermentation. The water cycle algorithm has been utilized for the modeling of two CPs, *E. coli* and *S. cerevisiae* cultivation, in [20].

The above-mentioned studies have shown that even though each of these metaheuristics is effective, the future belongs to a newer generation of algorithms known as hybrids. These hybrids combine and improve upon the strengths of several algorithms, while also overcoming their weaknesses [21,22].

Based on the results available so far, in this research, the authors seek a convenient metaheuristic algorithm to be hybridized with GA. An appropriate algorithm is the crow search algorithm (CSA) [15]. CSA is a metaheuristic algorithm based on bird swarm intelligence in searching for and hiding food. Because of its simplicity and ease of use, it has been widely adopted in various fields, such as image processing [23], task scheduling in cloud computing [24], economic load dispatch tasks [25] and other engineering problems [26]. The published results suggest that CSA is suitable for optimization problems and applicable for the identification of the model parameters of cultivation processes.

Since 2016, when Askarzadeh proposed CSA [15], many hybridizations, modifications and improvements to the algorithm have been developed to enhance its effectiveness and efficiency. The main parameters that influence the algorithm's performance and its search abilities have been studied recently in [27,28]. CSA has been reported to have a slow convergence speed and to fall prematurely into a local optimum, leading to numerous studies on CSA hybridization and modification. Some of the CSA improvements proposed lately are listed in Table 1, in addition to the utilized values of the CSA parameters, *fl* and *AP*. The parameters' values will be discussed later in the paper.

Algorithm improvement can be achieved by (1) integrative hybridization, combining the ideas of different metaheuristic principles or different optimization strategies, such as local search techniques; (2) collaborative hybridization, i.e., the sequential or parallel execution of different metaheuristic algorithms; and combinations of (1) and (2). Some examples of CSA modifications are discussed below.

Table 1. Recently proposed CSA hybridizations, modifications and improvements.

Algorithm	Reference, Year	Algorithm Parameters	
		Flight Length, *fl*	Awareness Probability, *AP*
Crow search algorithm (original)	[15], 2016	1.5, 2, 2.5	0, 0.05, 0.2
RCSA—CSA with the rough searching scheme	[2], 2018	dynamic	0.1
HCSA—CSA with Nawaz–Enscore–Ham heuristic technique and SA combined with the variable neighborhood search	[29], 2019	10	0.25
IFCSA—CSA based on improved flower pollination algorithm	[30], 2021	2	0.1
ImCSOA—CSA with chaotic maps	[31], 2021	1.8	0.1
MCSA—CSA with the innovative selection of the crows and adaptive adjustment of the flight length	[25], 2018	1.9, 2	0.1
CSA with modified awareness probability and random perturbation	[32], 2018	2	dynamic
CSA with dynamic *fl* and *AP*	[33], 2023	dynamic	dynamic
Improved CSA with dynamic *AP*	[34], 2020	2, 2.5	dynamic
ICSA—CSA improved by introducing experience factor, adaptive adjustment operator and Lévy flight distribution in position updating mechanism of crows	[35], 2017	2	0.1
ICSA—improved CSA with a new update mechanism	[36], 2021	1.2–1.6	0.1
Classical CSA using the VSA evolution mechanism to revise and exploit the solution space	[37], 2021	2	0.1, 0.5
Improved CSA with multi-strategy disturbance	[38], 2022	2	0.1
CSA with an improved objective function	[39], 2022	0.7	0.5
CSA with a cosine function and incorporating the opposition-based learning concept	[40], 2020	2	0.1
CSA with chaos and multiple opposition-based learning techniques	[41], 2023	2	0.1
Enhanced CSA with a free-fly mechanism and the personal upper-bound strategy	[42], 2019	1.5, 2, 2.5, 3	0, 0.1, 0.2
Chaotic CSA (CCSA)	[43], 2019	2	0.1
GWOCSA—hybrid grey wolf optimization with CSA	[44], 2019	2	0.1
HCSUC—hybrid CSA and uniform crossover algorithm	[45], 2021	2	0.1
CFCSA—hybrid CSA algorithm integrated with chaos theory and fuzzy c-means algorithm	[46], 2020	2	0.1
CCSA—hybrid cuckoo CSA	[47], 2021	2	0.1
Hybrid support vector regression and CSA	[48], 2022	2	0.1
Improved CSA based on arithmetic crossover	[49], 2022	2	0.1

Table 1. *Cont.*

Algorithm	Reference, Year	Algorithm Parameters	
		Flight Length, *fl*	Awareness Probability, *AP*
CSA with a particle swarm algorithm search strategy	[50], 2021	2	0.1
Crow search algorithm for efficient feature selection	[51], 2018	0.2	0.1
BCSA—hybrid binary CSA based on quasi-oppositional method	[52], 2023	1–1.8	0.2
MHCSA—memory-based hybrid CSA with particle swarm optimization algorithm	[53], 2023	-	-
CSA with implemented multi-strategy approach with a selection mechanism	[54], 2022	2	0.1
Enhanced chaotic crow search and particle swarm optimization algorithm	[55], 2021	2	0.2

Hassanien et al. [2] proposed a hybrid metaheuristic approach, RCSA, integrating CSA with the rough searching scheme (RSS). Combining the strengths of the two algorithms resulted in the enhanced exploration of auspicious regions. The work [29] presented a hybrid CSA (HCSA) using a Nawaz–Enscore–Ham approach to generate a diverse population with high quality. The problem of the premature convergence of the algorithm was tackled by simulated annealing, used additionally to improve the local search and the quality of the solutions. The paper [25] introduced a modified CSA (MCSA) with the innovative selection (priority-based selection) of the crows and the adaptive adjustment of the flight length. The superiority of the algorithms proposed in [2,25,29] was confirmed based on different well-known test systems and a set of benchmark test functions (BTFs). The arithmetic crossover concept in GA was incorporated into CSA to classify stages of brain tumors in [49]. Compared with the classical particle swarm optimization (PSO) and GA, CSA was found to be robust for standard BTFs. Clusters of similar users in recommendation systems were traced by a hybrid crow and a uniform crossover algorithm in [45]. The genetic crossover operator was built once again in CSA to increase the population diversity and to prevent the algorithm from trapping into a local optimum. A new update mechanism that increases CSA's convergence and improves its local search ability was proposed by Gholami et al. [36]. The advanced CSA was reported to have more stable convergence when optimizing the objective functions in terms of the solution accuracy and efficiency, thus having better results than the conventional algorithm and other metaheuristics. A different approach was found to improve the convergence accuracy and search ability of the algorithm in [38], by combining CSA with multi-strategy disturbance. The anticipated overall good performance was shown by testing the hybrid on twenty well-known BTFs. Durgut and Aydin [54] proposed a multi-strategy search for CSA. An improved CSA objective function combining the branch and predicate distance was used by Sharma and Pathik [39]. The time efficiency and better performance of the approach were shown for a test case generation problem. Hamed Alnaish and Algamal [52] developed a CSA based on a quasi-oppositional method. The presented results, such as high classification accuracy, a small number of selected features, and short computational time, confirmed the applicability of this novel algorithm in feature selection with classification. A cosine function and the opposition-based learning concept were incorporated into CSA in [40]. A free-fly mechanism in CSA was suggested for constraint handling [42]. The comparative performance of the above-mentioned algorithms over a large set of BTFs demonstrated that the enhanced approaches outperformed well-known competing methods.

Recently, chaos theory has emerged as a promising approach to enhance the performance of metaheuristic algorithms [31,46]. In [43], a chaotic crow search algorithm was presented to overcome CSA's low convergence rate and trapping in local optima. The sine chaotic map significantly improved CSA's performance. Gupta et al. [51] introduced a novel improved chaotic CSA. The performance of the algorithm, measured on 20 BTFs and compared with the original chaotic CSA, proved it to be more stable and accurate. A hybridization between chaotic CSA and PSO, capable of converging to the best global solution in the search field, was introduced by [55]. The comparison between the obtained fitness value and the standard deviation showed that the hybrid outperformed state-of-the-art methods. An enhanced CSA with chaos and multiple opposition-based learning techniques for global optimization problems was presented in [41]. The modification offered three levels of balancing global exploration and local exploitation.

Several collaborative CSA hybrids have been proposed in the last three years. Braik et al. [53] developed a memory-based hybrid of CSA and the PSO algorithm to reinforce the diversity of the population and to achieve robust search performance. A particle swarm search strategy built up the CSA global search capability to optimize the hidden layer neurons and connection weights of extreme learning machines in [50]. A new CSA hybrid scheme based on the flower pollination algorithm (FPA) was proposed by Cheng et al. [30] to tackle CSA's reported disadvantages. An improved FPA with a cross-pollination strategy and Cauchy mutation was exploited to hybridize CSA. A hybrid of the grey wolf optimizer (GWO) and CSA was presented in [44]. The strengths of both algorithms were merged in the GWOCSA hybrid to achieve global optima. A cuckoo CSA hybrid was introduced in [47] to optimize the task scheduling process in cloud computing. As in the cuckoo search algorithm, the crows grew capable of looking for a better food source among the neighbors and subsequently stealing the neighbors' food. Cortés-Caicedo et al. [37] used the evolution process of the vortex search algorithm to improve CSA. Applying a Gaussian probability distribution, they achieved better repeatability and a more effective algorithm. In [48], a hybrid between the support vector regression (SVR) technique and CSA was proposed. The statistically confirmed superior performance of the collaborative CSA hybrids was shown in their applicability to solve real-world complex problems using a set of BTFs.

Finally, to improve CSA's performance, some authors have applied dynamic algorithm parameters. In [33], the authors proposed dynamic AP and fl, whose values change at each iteration of the algorithm. These dynamic parameters improved the exploitation and exploration performance of the algorithm compared to a set of competing metaheuristic algorithms, not only in the case of both unimodal and multimodal BTFs but also for five real engineering problems. Wijayaningrum and Putriwijaya [34] also applied dynamic AP. The search agents' awareness probability was updated based on their most recent fitness value, balancing exploration and exploitation. Díaz et al. [32] presented an improved CSA with dynamically adjusted AP based on the current fitness value and Lévy flight movement incorporated to enhance the search process. Jain et al. [35] introduced a CSA with an improved balance between exploitation and exploration, including an experience factor, an adaptive adjustment operator, and a Lévy flight distribution when updating the crows' positions. The algorithm's performance, validated on high-dimensional nonlinear BTFs, was found superior to that of other well-established optimization algorithms.

While many improvements of the algorithm have been pointed out, there is still room for enhancements in terms of population diversity, balancing the local and global search abilities, and improving the convergence speed.

The present research discusses a hybrid metaheuristic algorithm that combines GA and CSA. CSA has not been studied and applied yet for parameter identification problems for nonlinear models of CPs. GA, on the other hand, is a metaheuristic algorithm highly applicable in solving such a complex problem. Thus far, it has been considered to adjust the position of the crows after a flight with the help of only one genetic operator—crossover [45,49]. The proposed GA-CSA hybrid fully integrates GA into CSA's initial group strategy, aiming to distribute the initial crows' population more uniformly and in a more focused manner. GA-CSA employs the good exploration abilities of GA to improve the convergence of the pure CSA. The expectation is that GA-CSA requires significantly fewer computational resources to obtain a more accurate model. The proposed hybrid is applied for the parameter identification procedure of a nonlinear model of the *E. coli BL21(DE3)pPhyt109* fed-batch cultivation process [56–58].

The GA-CSA results are compared in terms of model accuracy with the results of the pure CSA and GA, applied to the same problem, along with the two deterministic algorithms, SQP and the Q-N method. A mathematical approach designed to discover certain relationships between given criteria, namely InterCriteria analysis (ICrA) [59], is employed to confirm that there is a statistically significant difference between the obtained models. The ICrA approach has been extensively employed in a wide range of problems. Its applicability and correctness have been validated by the obtained results [60–64]. The ICrA results are further confirmed by performing parametric statistical tests, such as one-way analysis of variance (ANOVA) [65], and nonparametric test, such as Friedman [66] and Wilcoxon [67].

The main contributions of this study are as follows.

(1) A hybrid technique, GA-CSA, that combines the exploration abilities of GA and the exploitation of CSA is proposed.
(2) The GA-CSA hybrid is applied for the model parameter identification of the *E. coli BL21(DE3)pPhyt109* fed-batch cultivation process. The obtained mathematical model is successfully verified.
(3) The proposed hybrid model outperforms the pure GA and CSA methods in terms of accuracy while utilizing significantly fewer computational resources, such as computational time and memory. The resource usage is reduced by 8 to 10 times.
(4) The improved performance of the newly proposed hybrid GA-CSA, when compared to the pure CSA and GA in terms of model accuracy, is approved by applying ICrA and several classical statistical tests.

The rest of the paper is organized in the following manner. The mathematical model of the *E. coli BL21(DE3)pPhyt109* fed-batch cultivation process and the metaheuristic algorithms engaged in the parameter identification procedure are described in Section 2. The essence of the proposed GA-CSA hybrid algorithm is presented in Section 3. Section 4 presents and discusses the observed numerical results, obtained models, and additional investigations. Some concluding remarks are given in Section 5.

2. Materials and Methods

2.1. Escherichia coli BL21(DE3)pPhyt109 Fed-Batch Cultivation Process

2.1.1. Fed-Batch Cultivation Process

Fed-batch cultivation experiments for the extracellular production of bacterial phytases used the *E. coli* strain *BL21(DE3)pPhyt109.* Details of the experimental conditions and experimental data can be found in [56,58]. Only a brief description is provided below.

The experiments were performed in the Department of Fermentation Engineering, Faculty of Technology, University of Bielefeld. The gene of the *E. coli* phytase is contained in the plasmid pPhyt109, derived from the multi-copy plasmid pUC19 under the constitutive promoter of the *bglA* gene of *Bacillus amyloliquefaciens* [57].

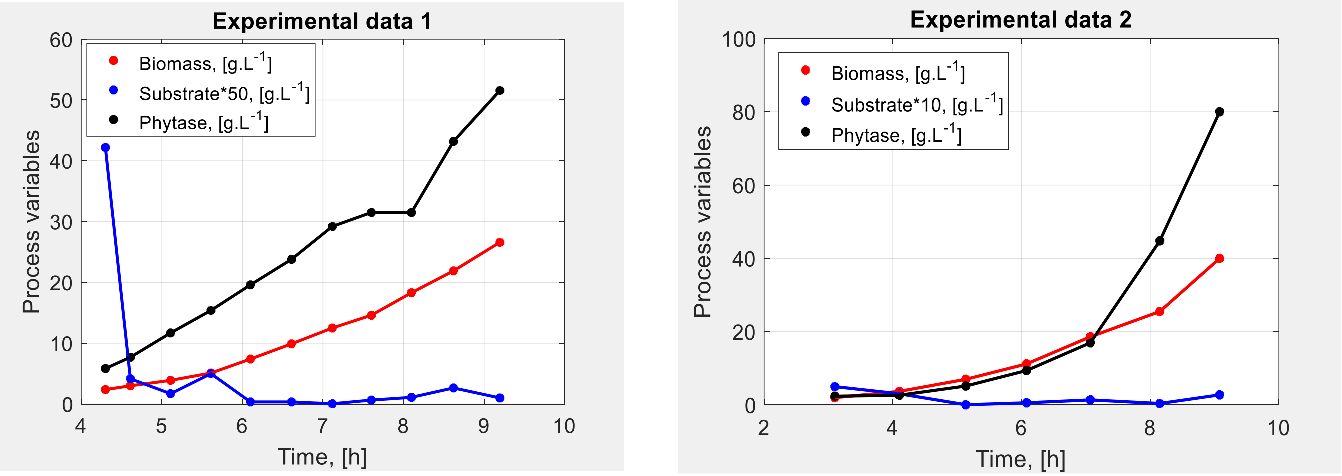
The present research considers two experimental data sets from two fed-batch cultivation processes. The cultivation conditions are presented in Table 2.

The first set of experimental data has been used for the identification of the model parameters (Figure 1a), while the second data set has been employed for the verification of the obtained mathematical model (Figure 1b).

Table 2. *Escherichia coli BL21(DE3)pPhyt109* fed-batch cultivation process conditions.

Condition	Cultivation Process 1	Cultivation Process 2
growth medium	glucose mineral salt medium	
bioreactor working volume	5 L	
bioreactor total volume	7 L	
temperature	37 °C	
airflow	10 L·min^{-1}	
stirrer speed	500 rpm	
pH	6.9 $^{\#}$	
t_0	4.30 h	3.10 h
$V(t_0)$	2.70 L	
$\gamma_X(t_0)$	3.20 g/L	
$\gamma_S(t_0)$	0.78 g/L	0.50 g/L
$\gamma_{S_{in}}$	500 g/L	
$\gamma_{S_{sp}}$	0.2 g/L	0.1 g/L

$^{\#}$ by controlled addition of 4 N NaOH antifoam (PE8100, BASF, Germany).



(**a**) (**b**)

Figure 1. Time profiles of the process variables—real experimental data: (**a**) data used for modeling; (**b**) data used for model verification.

2.1.2. Mathematical Model of *E. coli BL21(DE3)pPhyt109* Fed-Batch Cultivation Process

The mathematical model is based on the following assumptions.

- To simplify the model, all possible effects of mixing the highly concentrated feeds with the cultivation medium are ignored. The bioreactor is completely mixed.
- Throughout the experiment, the viscosity of the suspension in the reactor remains constant.
- Biomass, phytase, and water are the main products of *E. coli* cultivation.
- The substrate (glucose) is consumed mainly oxidatively.
- The growth conditions are balanced.
- Balanced growth conditions are assumed, meaning that any deviation in the growth rate, substrate consumption, or phytase production is not expected to have a significant impact on the elemental composition of the biomass.
- The production of phytase, for simplicity, is considered to be a one-step enzymatic reaction.

The fed-batch cultivation of *E. coli BL21(DE3)pPhyt109* involves the measurement of rates associated with cell growth, sugar consumption, and phytase production. These rates are commonly evaluated through the use of a mass balance as follows [56]:

$$\frac{d\gamma_X}{dt} = \mu_{max}\frac{\gamma_S}{k_S+\gamma_S}\gamma_X - \frac{Q}{V}\gamma_X, \tag{1}$$

$$\frac{d\gamma_S}{dt} = \frac{1}{Y_{S/X}}\mu_{max}\frac{\gamma_S}{k_S+\gamma_S}\gamma_X + \frac{Q}{V}(\gamma_{Sin}-\gamma_S), \tag{2}$$

$$\frac{d\gamma_P}{dt} = \frac{1}{Y_{P/X}}\mu_{max}\frac{\gamma_S}{k_S+\gamma_S}\gamma_X - \frac{Q}{V}\gamma_P, \tag{3}$$

$$\frac{dV}{dt} = Q, \tag{4}$$

where γ_X—concentration of the biomass, g/L; γ_S—concentration of the substrate (glucose), g/L; γ_P—concentration of the phytase, g/L; Q—influent flow rate, h^{-1}; V—bioreactor volume, L; γ_{Sin}—influent glucose concentration, g/L; μ_{max}—maximum specific growth rate, h^{-1}; $Y_{S/X}$ and $Y_{P/X}$—yield coefficients, g/g; k_S—saturation constant, g/L.

Four model parameters should be identified—$p = [\mu_{max}\ k_S\ Y_{S/X}\ Y_{P/X}]$.

The parameter identification problem is essentially minimizing the distance measure J between the experimental data and the values of the state variables predicted by the model.

The model parameter's vector p was estimated based on the objective function J:

$$J = \sum_{i=1}^{n}\left\{\left[\gamma_{Xexp}(i) - \gamma_{Xmod}(i)\right]\right\}^2 + \left\{\left[\gamma_{Sexp}(i) - \gamma_{Smod}(i)\right]\right\}^2 + \left\{\left[\gamma_{Pexp}(i) - \gamma_{Pmod}(i)\right]\right\}^2 \rightarrow min, \tag{5}$$

where n is the length of the data vector for the state variables γ_X, γ_S, and γ_P; γ_{Xexp}, γ_{Sexp}, and γ_{Pexp} are the existing experimental data; γ_{Xmod}, γ_{Smod}, and γ_{Pmod} are the model predictions for a given set of parameters.

2.2. Metaheuristic Algorithms for Model Parameter Identification

2.2.1. Crow Search Algorithm

The crow search algorithm (CSA), introduced by Askarzadeh in 2006 [15], depicts the intelligent behavior of crows.

Crows live in flocks. Each crow in the flock stores food in a secret place and can retrieve it months later if necessary. The birds' intelligence is expressed in many ways, including how they observe and follow other crows to their hidden places and steal the food buried there. If a crow senses that it is being followed, it tries to deceive the pursuer.

How is the crows' behavior simulated by the algorithm for a specific problem?

The number of crows in the flock is N. The current position of each crow is presented as a d-dimensional vector, $crow_i = \left\{crow_i^1, \ldots, crow_i^d\right\}$, $i \in [1; N]$, where d is the problem size. Each position is a feasible solution to the considered problem. Initially, all crows in the flock are positioned randomly. Each crow has the position of the hiding place stored in its memory, mem_i. This represents the best position of the crow yet.

The crows are in constant search of better food sources. At a certain point in time, $crow_i$ may follow $crow_j$. If $crow_j$ is unaware that it is being followed, it will lead $crow_i$ to its hiding place. The current position of $crow_i$ is adjusted accordingly:

$$crow_i = crow_i + r_i \times flight\_length \times \left(mem_j - crow_i\right), \tag{6}$$

where $r_i \in [0; 1]$ is a random number with a uniform distribution. However, if $crow_j$ is aware of the presence of $crow_i$, in order to protect its hiding place, it will try to deceive the pursuer by taking it somewhere random. The position of $crow_i$ is then changed to a randomly generated one. The awareness of a crow is modeled by a random number and compared to an awareness probability parameter.

CSA depends on a few parameters: the number of crows in the flock N, the maximum number of iterations $MaxIter$, the flight length fl, and the awareness probability AP.

2.2.2. Genetic Algorithm

The genetic algorithm (GA), introduced by Holland [10,11], is an abstraction of Darwin's evolution theory. Generation after generation, individuals try to adapt to environmental changes in order to survive and reproduce.

How does the population evolve through the generations?

The size of the population is $NInd$. For a problem of size d, each individual is encoded as a d-dimensional vector, an analogue of a chromosome and a possible solution. Their ability to adapt is modeled by a fitness function. Since the population size is constant, some individuals die and are replaced by new ones.

The next generation is formed by selected individuals on which genetic operators such as crossover and mutation are applied. Individuals with better fitness evaluations are more likely to reproduce. Parts from the selected individuals are combined. Therefore, the newly generated solutions share similar characteristics with their parents. The mutation of an individual is achieved by changing randomly selected parts.

The genetic operators aim to take advantage of the knowledge acquired over generations but also to ensure diversity in the population.

GA depends on more input parameters compared to CSA: the population size $NInd$, the maximum number of generations $MaxGen$, the generation gap $GGAP$, the crossover probability p_c and the crossover operator, the mutation rate p_m and the type of mutation, and the criterion for selecting the best individuals.

3. Hybrid GA-CSA

CSA has the potential to find the optimum solution for certain configurations of the search space [15,68]. Randomly generated initial solutions affect the convergence speed of the algorithm and the required computational resources. The more poorly distributed in the search space are these initial solutions and the further they are from the sought optimal solution, the longer it takes for the algorithm to reach this optimum.

A collaboration between two algorithms, GSA and GA, is hereby investigated in an attempt to study the algorithms' behavior, minimize their drawbacks, and accelerate their advantages.

GA is designed to explore a wide range of the search space. However, it converges slowly [69,70]. In the proposed GA-CSA hybrid, GA is employed in the CSA initialization phase to generate the crows' initial positions. For just a few generations, these good exploration abilities of GA are exploited to guide the CSA initial population in the correct direction. Providing a good starting point, CSA requires a significantly reduced population size and number of iterations to reach the sought optimum (or a close enough) solution. The computational resources necessary to improve the solution accuracy are also essentially decreased.

The pseudo-code of the proposed GA-CSA hybrid algorithm is presented below (Algorithm 1).

Algorithm 1: Pseudo-code of the hybrid GA-CSA

```
Begin
  define the GA input parameters: GA operators, NInd, MaxGen, GGAP, $p_c$, and $p_m$
  define the CSA input parameters: N, MaxIter, fl, AP
  problem initialization: number of parameters d, parameters' bounds,
  objective function f(x), process model, experimental data
  % initialization phase of CSA
  for i := 1 to N
    generate randomly NInd number of individuals
    evaluate the individuals in the population
    for j := 1 to MaxGen
      select individuals from the current generation
      perform crossover on the selected individuals with a probability $p_c$
      perform mutation on each individual with a probability $p_m$
      place the offspring into the new population
      evaluate the individuals in the new population
    end for
    rank the individuals in the population
    store the best individual and its estimation
  end for
  % intrinsic part of CSA
  initialize the memory of each crow
  for iter := 1 to MaxIter
    for i := 1 to N (all crows in the flock)
      choose randomly a crow to follow
      define an awareness probability $r_i$
      if $r_i \geq AP$
        change the current position of the $crow_i$
      Else
        generate a new random position of the $crow_i$
      end if
    end for
    check if all new positions are feasible
    evaluate the new positions
    update the memory of each crow
  end for
  rank the position of the crows in the flock
  store the best position
End
```

4. Numerical Results and Discussion

4.1. Parameters' Algorithms Tuning

GA has been applied repeatedly for the mathematical modeling of cultivation processes [9,71,72]. Based on the thorough investigation of the parameters' influence [8], the parameters' values and GA functions have been set as follows:

- the population size *NInd* = 100 (pure GA) and *NInd* = 25 (hybrid GA-CSA);
- the maximum number of generations *MaxGen* = 100 (pure GA) and *MaxGen* = 25 (hybrid GA-CSA);
- the generation gap *GGAP* = 0.97;
- the crossover probability $p_c = 0.7$;
- the crossover operator—extended intermediate recombination;
- the mutation rate $p_m = 0.1$;
- the type of mutation—real-value mutation like Breeder genetic algorithm [73];
- the selection operator—roulette wheel selection.

Since CSA has been applied for the first time for the parameter identification of the mathematical model of the considered cultivation process, the initial algorithm's parameters have been chosen based on values known in the literature (Table 1) and several pre-tests. The investigated values are fl = [1 2 3] and AP = [0.01 0.05 0.1 0.2 0.3]. The following CSA parameters have been selected:

- the population size N = 100 (pure CSA) and N = 25 (hybrid GA-CSA);
- the maximum number of iterations $MaxIter$ = 100 (pure CSA) and $MaxIter$ = 50 (hybrid GA-CSA);
- the flight length fl = 2;
- the awareness probability AP = 0.1.

In the hybrid algorithm, a small GA population was engaged in the beginning. The set of initial solutions for CSA was generated for only 25 iterations. The population of the same size evolved over 50 iterations.

4.2. Parameter Identification of E. coli BL21(DE3)pPhyt109 Fed-Batch Cultivation Process

The parameter identification of the nonlinear mathematical model (Equations (1)–(4)) of the *E. coli BL21(DE3)pPhyt109* fed-batch cultivation process performed using the GA-CSA hybrid algorithm is described below.

4.2.1. Simulation Setup

The proposed mathematical model consists of a set of four ODEs (Equations (1)–(4)) with three dependent state variables $x = [\gamma_X\ \gamma_S\ \gamma_P]$ and four unknown parameters $p = [\mu_{max}\ k_S\ Y_{S/X}\ \ Y_{P/X}]$.

A set of individuals that were potential solutions to the problem at hand formed the population of the GA-CSA hybrid. The solutions were represented as vectors whose elements corresponded to the unknown model parameters. Based on the authors' expertise [56], the ranges of these parameters were estimated as the following:

$$0 \le \mu_{max} \le 0.9;\ 0.03 \le k_S \le 0.5;\ 0.5 \le Y_{S/X} \le 10;\ 0.5 \le Y_{P/X} \le 10. \quad (7)$$

The numerical experiments were performed on an Intel® Core™i7-8700 CPU @ 3.20 GHz, 3192 MHz, 32 GB Memory (RAM), with a Windows 10 pro (64 bit) operating system. The considered competing algorithms were implemented in Matlab R2019a. The implementation of the GA-CSA hybrid technique followed the above-presented pseudocode. The mathematical model of *E. coli* was created in the Simulink R2019a environment. The solver options were the fixed step size of 0.01 and *ode4* (Runge–Kutta) with TIMESPAN = [0 10].

A series of parameter identification procedures of the described model (Equations (1)–(4)) were performed using GA-CSA. Due to the stochastic characteristics of the applied algorithms, 30 runs were conducted.

4.2.2. Numerical Results

The results of the application of the proposed GA-CSA hybrid algorithm to the parameter identification of the *E. coli BL21(DE3)pPhyt109* cultivation process model are presented in Tables 3–5. Table 3 summarizes the mean, the best, and the worst results for the criterion value J of the performed 30 runs and the obtained SD. The results obtained by the deterministic algorithms for different initial solutions are presented in Table 4. Since deterministic methods produce the same results for a particular set of inputs, a set of different initial solutions was tested. The results for the four best initial solutions are given in Table 4. Table 5 presents the estimated model parameters of all algorithms and the corresponding SD for the stochastic algorithms.

Table 3. Comparison of the objective function values obtained by GA-CSA and GA and CSA.

Algorithm	Objective Function, J			
	Mean	Worst	Best	SD
GA	121.0858	121.1001	121.0831	0.003546
CSA	120.1905	120.2070	120.1731	0.009986
GA-CSA	120.2095	120.3507	120.1724	0.032704

Table 4. Comparison of the objective function values obtained by SQP and Q-N for different initial solutions.

Algorithm	Initial Solution $[\mu_{max}; k_S; Y_{S/X}; Y_{P/X}]$	Objective Function, J
SQP Q-N	[0.85; 0.03; 2.5; 2.5]	225.3851 185.2192
SQP Q-N	[0.75; 0.08; 3.5; 3.5]	122.7380 483.1404
SQP Q-N	[0.8; 0.03; 2; 2.5]	227.5362 860.9834
SQP Q-N	[0.6; 0.05; 3.5; 3.5]	148.5591 378.2660

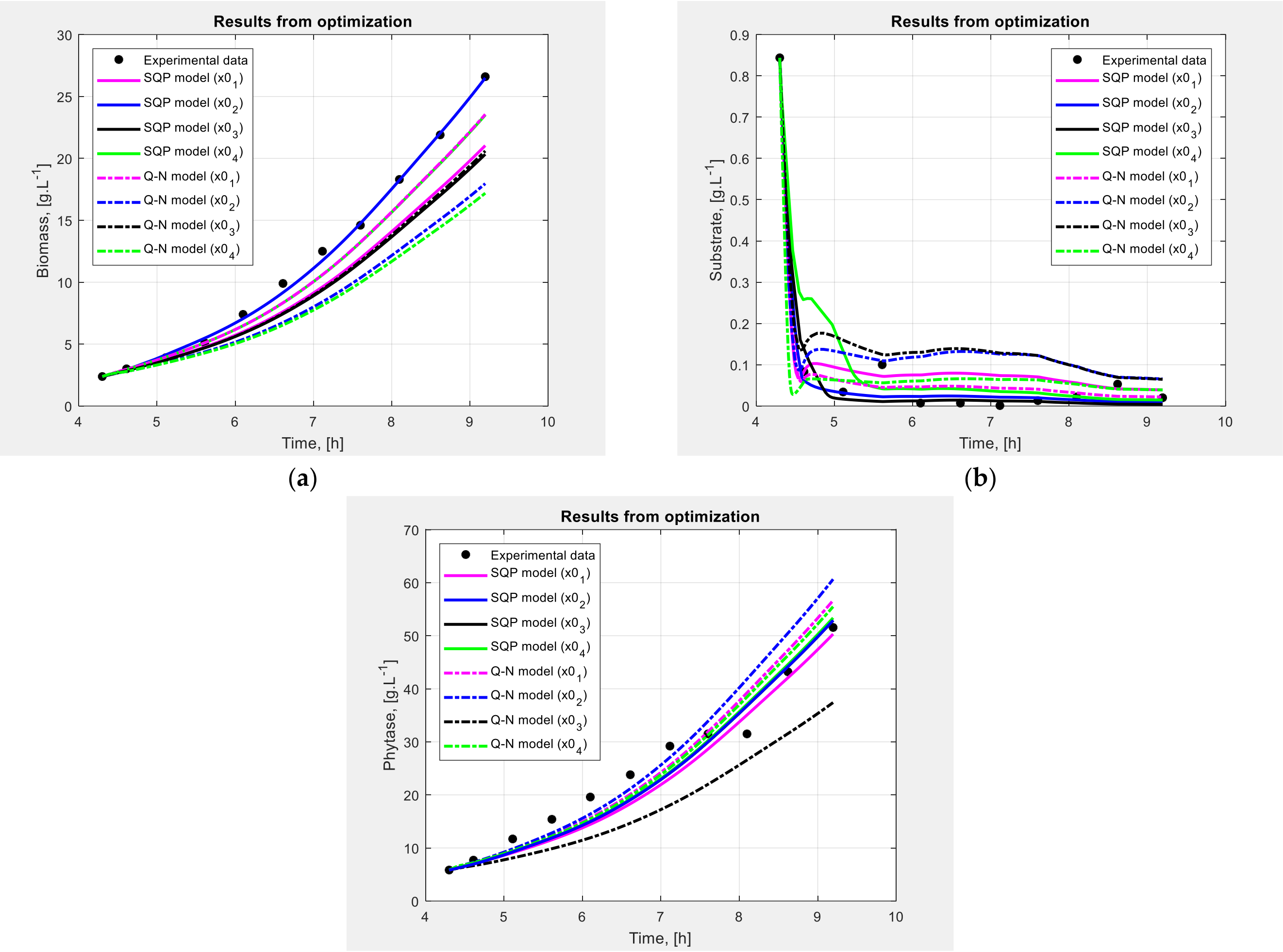
Table 5. Comparison of the obtained parameter estimates.

Algorithm	Model Parameter Estimates							
	μ_{max}, [h^{-1}]	SD	k_S, [g·L^{-1}]	SD	$Y_{S/X}$, [g·g^{-1}]	SD	$Y_{P/X}$, [g·g^{-1}]	SD
SQP	0.729	--	0.0156	--	2.276	--	1.957	--
Q-N	0.850	--	0.0315	--	2.588	--	2.400	--
GA	0.900	3.76×10^{-5}	0.0060	1.18E-06	2.262	0.0025	1.943	0.0025
CSA	0.888	0.0196	0.0054	0.00028	2.250	0.0029	1.943	0.0035
GA-CSA	0.892	0.0232	0.0054	0.00033	2.251	0.0036	1.944	0.0039

The results of the novel hybrid metaheuristic algorithm were compared to those obtained by the pure CA and CSA and the two deterministic methods—SQP and Q-N. All algorithms were applied for the considered problem with the same upper and lower parameters' bounds (Equation (7)).

The graphical results are presented in Figures 2 and 3. The time profiles of the process variables obtained by the QSP and Q-N models based on the selected best four initial solutions are presented in Figure 2. The best QSP and Q-N models are compared to the obtained GA, CSA, and hybrid GA-CSA models in Figure 3.

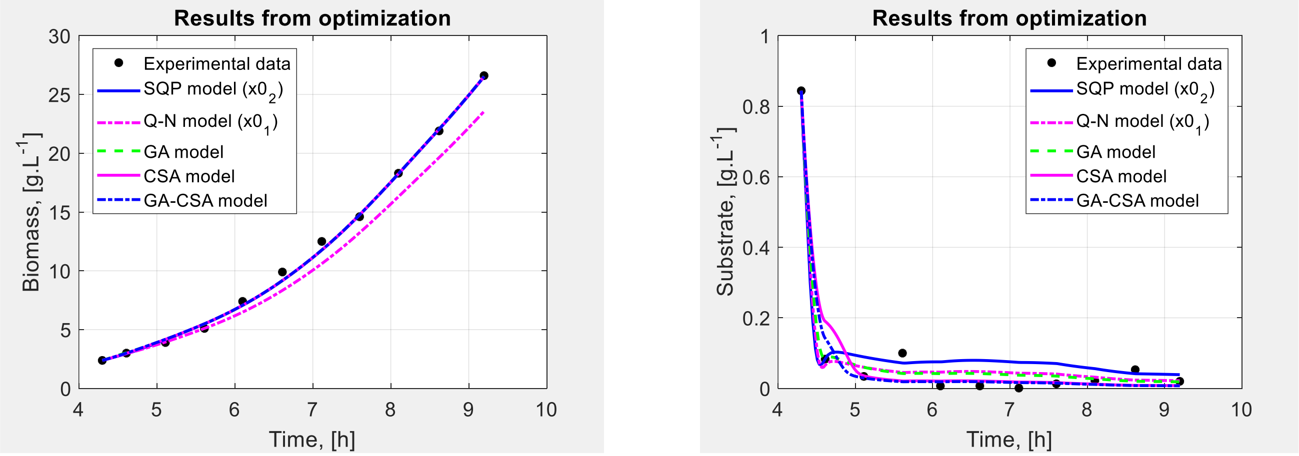
The proposed hybrid metaheuristic algorithm is capable of finding the best solution with a significantly smaller computational resource requirement. For instance, it requires a much smaller number of iterations (only 25 in the beginning and then 50) and a smaller population size (only 25% of the population size of GA and CSA). For example, for 100 individuals and 100 iterations, there are 100 × 100 = 10,000 objective function evaluations. For the GA-CSA hybrid, this number is 25 × 25 + 25 × 50 = 1875. By applying the hybrid GA-CSA, better results were obtained using 80% less computational resources. The results of QSP and Q-N were obtained based on approximately 1000 objective function evaluations. However, the strong dependence of the solution on the initial conditions makes it very difficult to use these methods, especially in cases where initial estimates cannot be set with good accuracy. As can be seen from Table 4, the initial solutions are quite close to those achieved by the metaheuristic algorithms. Only for these particular initial conditions, QSP and Q-N were able to yield satisfactory outcomes.



(**a**) (**b**)

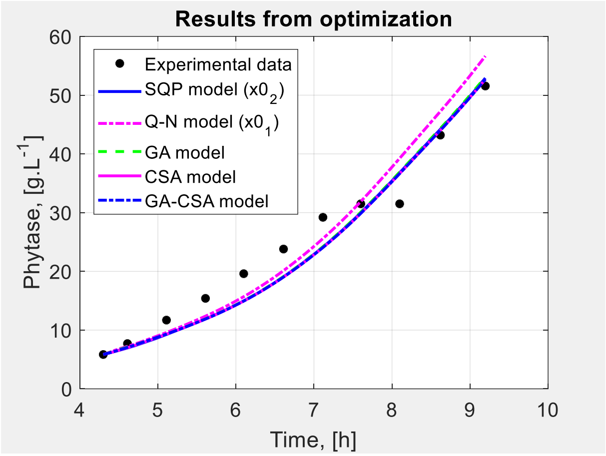
(**c**)

Figure 2. Deterministic model simulation: time profiles of the process variables—real experimental data and model-predicted data: (**a**) biomass γ_X; (**b**) substrate γ_S; and (**c**) phytase γ_P.



(**a**) (**b**)

Figure 3. *Cont.*



(**c**)

Figure 3. All models' simulation: time profiles of the process variables—real experimental data and model-predicted data: (**a**) biomass γ_X; (**b**) substrate γ_S; and (**c**) phytase γ_P.

CSA achieved the best mean value. Both the GA and CSA estimates had a better SD compared to GA-CSA. To further improve the hybrid algorithm, additional investigations are necessary to enhance the desired population distribution through hybridization. One possible direction is to apply adaptive CSA parameters (*fl* and *AP*), which will affect the exploitation and exploration performance of the algorithm [32–34].

In the next two sections, the obtained results are interpreted to show that they are statistically different, meaning that the considered metaheuristic algorithms have produced distinguishable models.

4.2.3. Interpretation of the Results

The next results represent the InterCriteria analysis and some statistical analyses performed over the results of the metaheuristic algorithms due to their stochastic nature.

InterCriteria Analysis of the Results

Firstly, to assess the similarity between the developed models, the ICrA approach was applied to evaluate the proximity of the GA, CSA, and hybrid GA-CSA results. An intuitionistic fuzzy pair (IFP) [74,75] was obtained as an estimation of the degrees of "agreement" (consonance) and "disagreement" (dissonance) between the two algorithms. The IFP is an ordered pair of real non-negative numbers $\left\langle \mu_{C,C'}, \nu_{C,C'} \right\rangle$ such that $\mu_{C,C'} + \nu_{C,C'} \leq 1$. The difference $\pi_{C,C'} = 1 - \mu_{C,C'} + \nu_{C,C'}$ is considered as a degree of "uncertainty" [76].

The analysis was carried out using the cross-platform software ICrAData v2.5 [77]. The calculations employed numerical data from 30 runs of each algorithm (GA, CSA, and GA-CSA) for the model parameter estimates and the values of the objective function. The resulting degree of agreement $\mu_{C,C'}$ and degree of disagreement $\nu_{C,C'}$ between the three considered algorithms are presented as index matrices [78] in the following form:

$$\begin{array}{cccc} \left\langle \boldsymbol{\mu_{C,C'}}, \mathbf{v_{C,C'}} \right\rangle & \text{GA} & \text{CSA} & \text{GA-CSA} \\ \text{GA} & \langle \mathbf{1}, \mathbf{0} \rangle & \langle 0.49, 0.51 \rangle & \langle 0.53, 0.47 \rangle \\ \text{CSA} & \langle 0.49, 0.51 \rangle & \langle \mathbf{1}, \mathbf{0} \rangle & \langle 0.71, 0.29 \rangle \\ \text{GA-CSA} & \langle 0.53, 0.47 \rangle & \langle 0.71, 0.29 \rangle & \langle \mathbf{1}, \mathbf{0} \rangle \end{array} \quad (8)$$

The obtained results are visualized in Figure 4 in the intuitionistic fuzzy interpretation triangle.

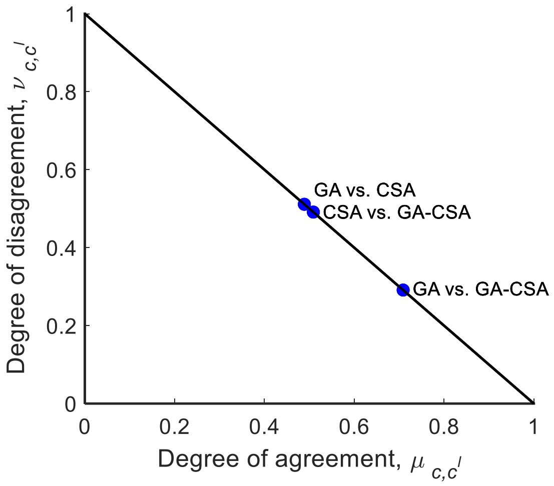


Figure 4. Representation of the results in the intuitionistic fuzzy interpretation triangle.

Each IF pair in Equation (8) estimates the similarity of the observed numerical results. The values defining the consonance and dissonance between the algorithms have been interpreted according to the scale defined in [76]. The obtained $\langle \mu_{C,C'}, \nu_{C,C'} \rangle$ for each pair of algorithms show that the numerical results are distinguishable. The pairs GA vs. CSA and GA vs. GA-CSA are in strong dissonance, i.e., the data do not show any correlation. The pair CSA vs. GA-CSA is in weak dissonance, i.e., there is weaker independence of the results compared to the other two pairs of algorithms. This result confirms the success of the collaborative hybridization of GA and CSA.

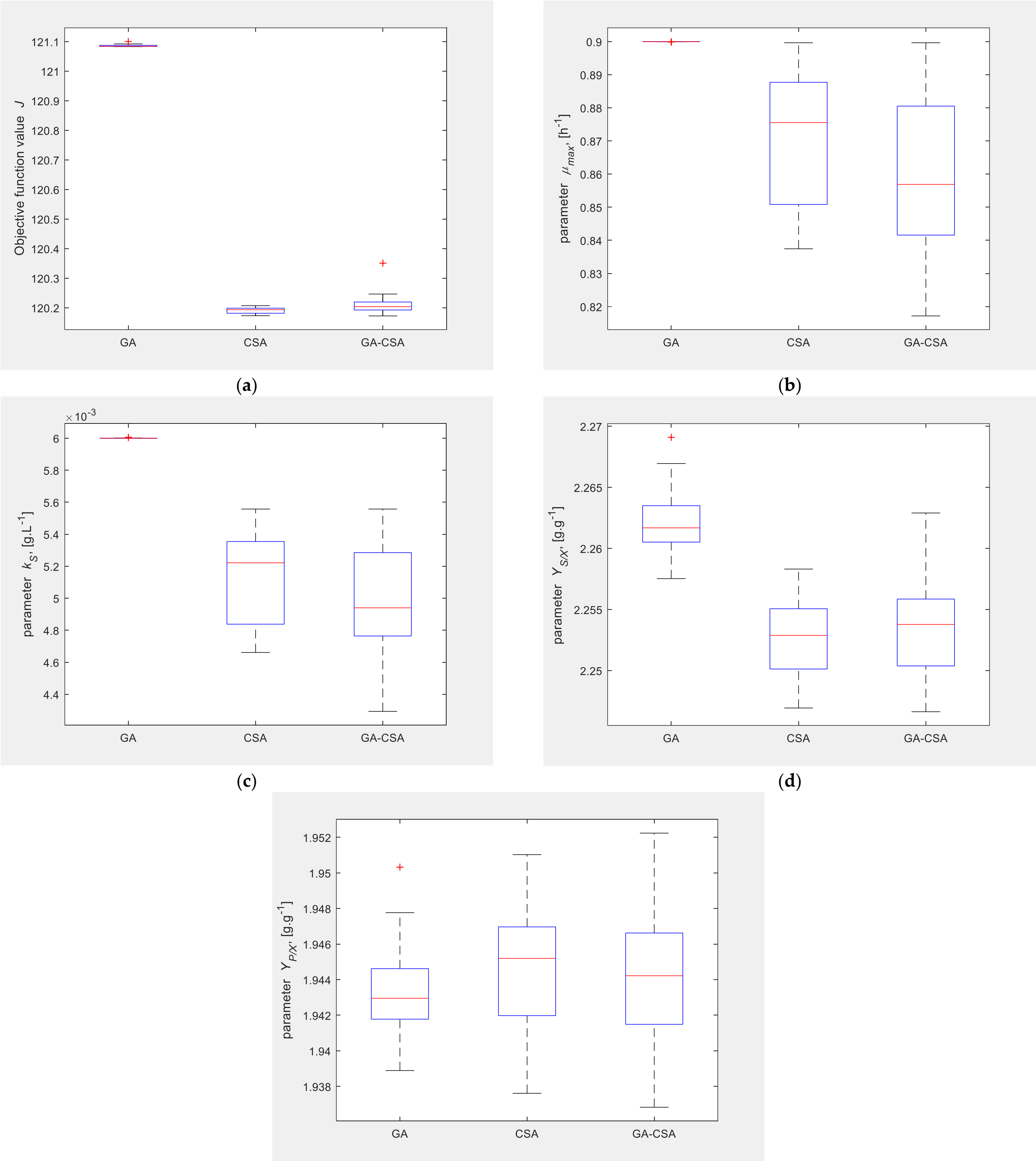
The strong dissonance between the results of GA and GA-CSA shows that the proposed hybrid has completely different behavior compared to GA. This indicates that the algorithm's exploration is substantially influenced by engaging GA in the initial group strategy, while CSA's exploitation is effectively maintained in the hybrid. Considering also the best objective function value achieved by the hybrid, it can be concluded that the aim of constructing a more uniform and focused distribution of the initial crows' population is fulfilled. The weaker independence between CSA and the hybrid shows that the behavior of the new algorithm is changed compared to the pure CSA, but, at the same time, the essential characteristics of CSA can still be found in the hybrid.

Statistical Analysis of the Results

The statistical analysis was performed on the same numerical data from 30 runs of the three algorithms. Box plot diagrams visualizing the summary statistics of the GA, CSA, and GA-CSA results (mean values, SD, and the median of the estimated model parameters and the obtained objective function value J) are presented in Figure 5.

Nonparametric tests, namely the Friedman test, the Wilcoxon test, and the one-way analysis of variance (ANOVA) test, were applied to compare the performance of the algorithms based on the objective function J evaluations obtained by the 30 runs of each algorithm. The results are presented in Figures 6 and 7 and Table 6.

The conducted statistical tests show that the results achieved by the pure GA and CSA are statistically different from those of the hybrid GA-CSA. In all performed tests, p-values substantially lower than 0.05 are observed. This indicates that the mean of the objective function is not the same for all three algorithms, and the obtained models are statistically distinguishable. These facts are confirmed by ICrA as well.

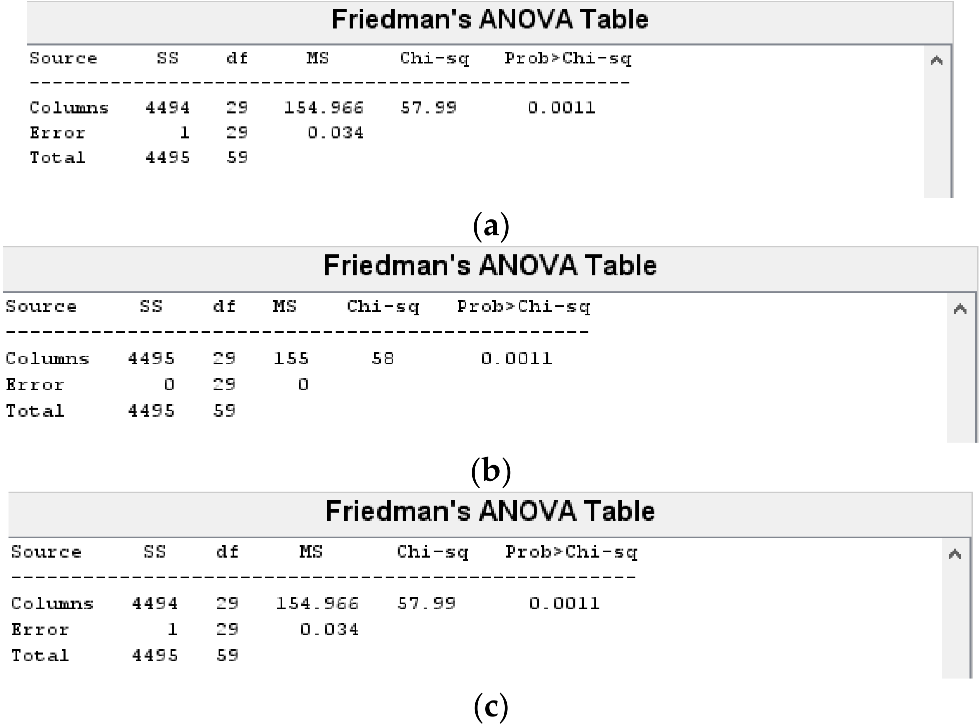


(**a**) (**b**)

(**c**) (**d**)

(**e**)

Figure 5. Box plot with the results from the parameter identification of an *E. coli BL21(DE3)pPhyt109* cultivation model: (**a**) objective function J; (**b**) parameter μ_{max}; (**c**) parameter k_S; (**d**) parameter $Y_{S/X}$; and (**e**) parameter $Y_{P/X}$.



```
Friedman's ANOVA Table
Source      SS     df      MS       Chi-sq   Prob>Chi-sq
-----------------------------------------------------
Columns    4494    29   154.966    57.99      0.0011
Error         1    29     0.034
Total      4495    59
```

(**a**)

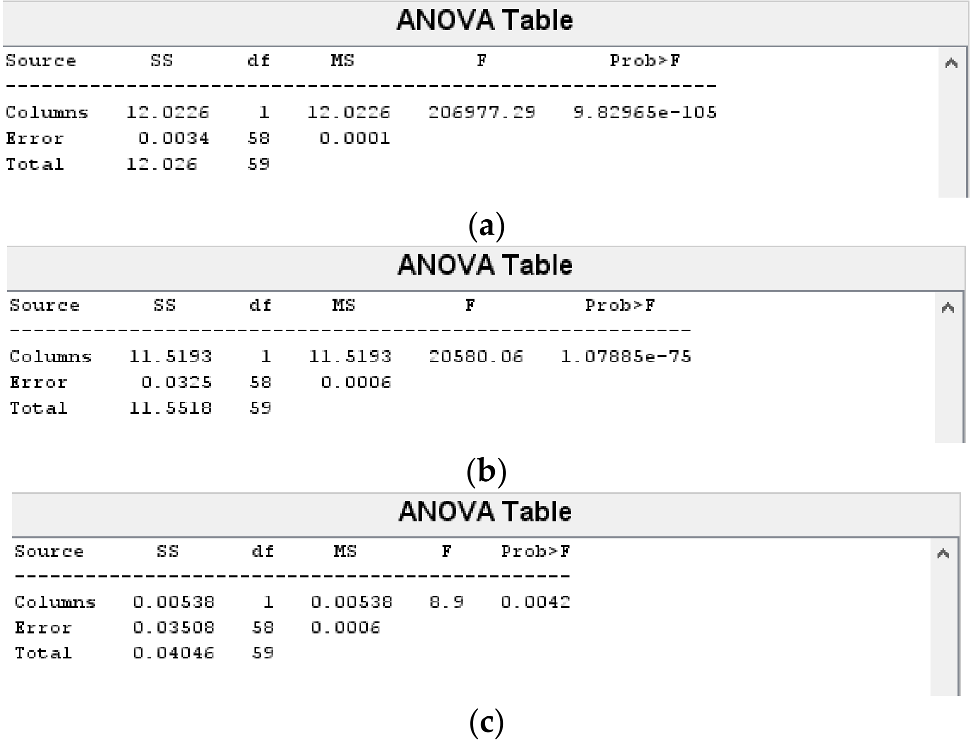
```
Friedman's ANOVA Table
Source      SS     df    MS     Chi-sq   Prob>Chi-sq
-----------------------------------------------------
Columns    4495    29   155       58        0.0011
Error         0    29     0
Total      4495    59
```

(**b**)

```
Friedman's ANOVA Table
Source      SS     df      MS       Chi-sq   Prob>Chi-sq
-----------------------------------------------------
Columns    4494    29   154.966    57.99      0.0011
Error         1    29     0.034
Total      4495    59
```

(**c**)

Figure 6. Statistical analysis—Friedman test: (**a**) GA vs. CSA; (**b**) GA vs. GA-CSA; and (**c**) CSA vs. GA-CSA.



```
ANOVA Table
Source       SS       df      MS            F            Prob>F
---------------------------------------------------------------
Columns    12.0226     1    12.0226    206977.29    9.82965e-105
Error       0.0034    58     0.0001
Total      12.026     59
```

(**a**)

```
ANOVA Table
Source       SS       df      MS           F          Prob>F
------------------------------------------------------------
Columns    11.5193     1    11.5193    20580.06    1.07885e-75
Error       0.0325    58     0.0006
Total      11.5518    59
```

(**b**)

```
ANOVA Table
Source       SS       df      MS        F     Prob>F
------------------------------------------------
Columns    0.00538     1    0.00538    8.9    0.0042
Error      0.03508    58    0.0006
Total      0.04046    59
```

(**c**)

Figure 7. Statistical analysis—ANOVA: (**a**) GA vs. CSA; (**b**) GA vs. GA-CSA; and (**c**) CSA vs. GA-CSA.

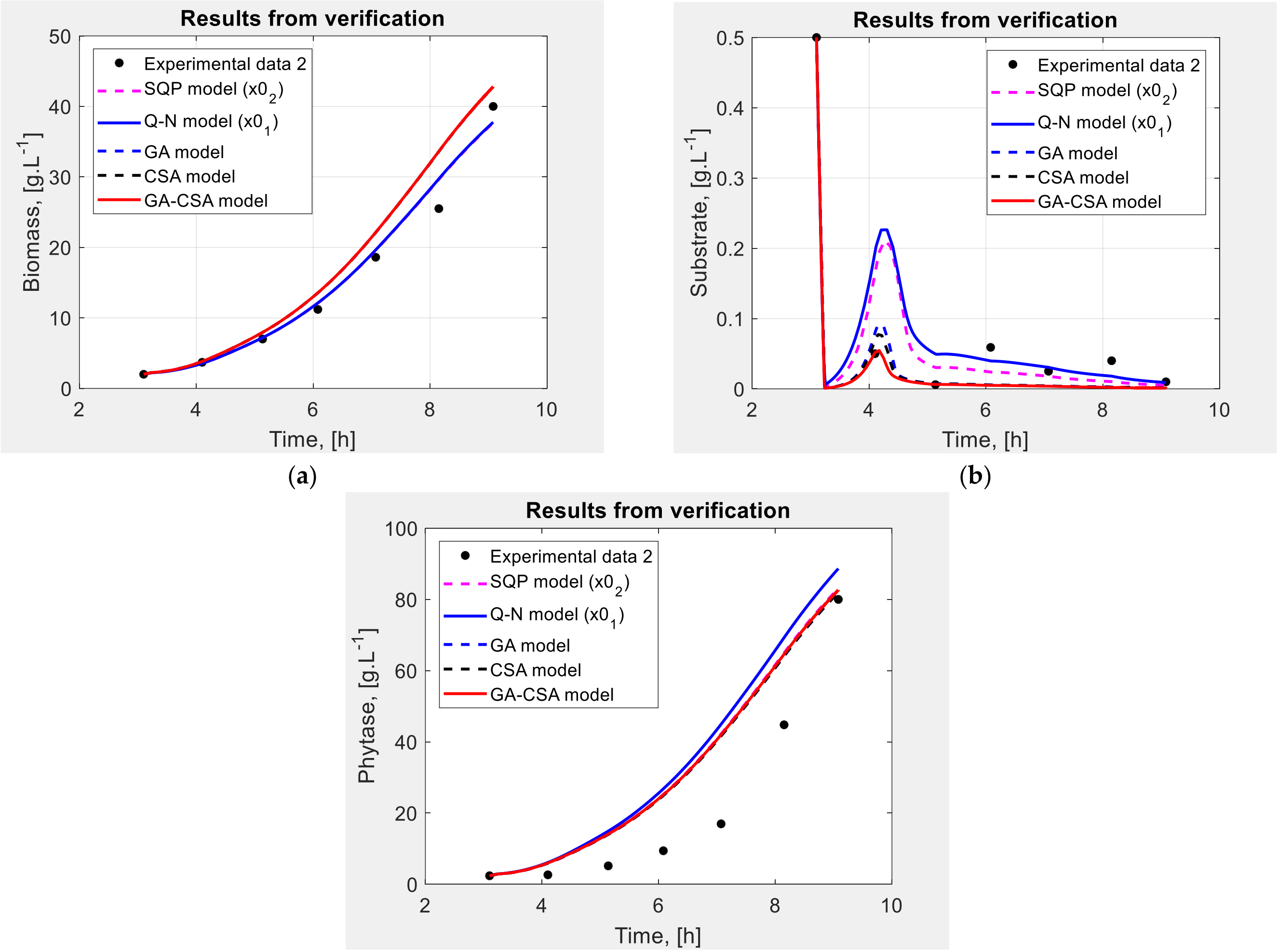
Table 6. Statistical analysis—Wilcoxon test.

Wilcoxon Test				
Algorithms	***p*-Value**	**H**	**STATS**	
			Zval	**Ranksum**
GA vs. CSA	3.0199×10^{-11}	1	6.6456	1365
GA vs. GA-CSA	2.9392×10^{-11}	1	5.5268	1451
CSA vs. GA-CSA	0.0047	1	−2.8244	723.5

4.3. Verification of the Obtained Mathematical Model of E. coli BL21(DE3)pPhyt109 Fed-Batch Cultivation Process

The developed mathematical models of the *E. coli BL21(DE3)pPhyt109* fed-batch cultivation process were verified with an independent set of experimental data. The characteristics of the cultivation are presented in Table 2 and Figure 1b.

The dynamics of the main process variables (biomass γ_X, substrate γ_S, and phytase γ_P) were compared to the real data set. The results are presented in Figure 8.



(**a**) (**b**)

(**c**)

Figure 8. Model verification: time profiles of the process variables—real experimental data and model predicted data: (**a**) biomass γ_X; (**b**) substrate γ_S; and (**c**) phytase γ_P.

To compare the quality of the proposed models, residuals, i.e., the differences between the experimental data and model-predicted values of data, were calculated based on Equation (5). The obtained J values (errors) are presented in Table 7.

Table 7. Model verification comparison of the objective function values obtained by SQP and Q-N at different initial solutions.

Algorithm	**Error, *J***
SQP	1620,2603
Q-N	1892,2863
GA	1408,1104
CSA	1408,0141
GA-CSA	1386,9798

The verification results are very positive. The developed models follow the behavior of the CP dynamics, with slight deviations in the description of the phytase. However, the models still exhibit the same type of behavior of the product change during the process (Figure 8c).

The results obtained for the description of the glucose concentration indicate rather unreliable data (noise, inaccurate analysis, wrong measurements, etc.). Some of the experimental data (Figure 8b) may be incorrect.

The observed deviations can be explained by the fact that the models were developed based on data from a CP where glucose was maintained at 0.2 $g{\cdot}L^{-1}$, while the verification data came from a CP with the glucose concentration set at 0.1 $g{\cdot}L^{-1}$.

The behavior of the models can be distinguished most clearly from the results shown in Figure 8b. The experimental data on glucose variation are best described by the hybrid GA-CSA, followed by CSA and GA. Considering the deterministic algorithms, the SQP model shows better performance than the Q-N method. The Q-N model seems to be the only one that exhibits distinct behavior for the process variables γ_X and γ_P, while the other five models predict these variables in an almost identical manner.

The results presented in Table 7 confirm the observation made through the graphical results from the model verification depicted in Figure 8. The estimated model residuals show that the GA-CSA hybrid model has the highest quality, indicated by the smallest error. The models estimated by GA and CSA have very close residuals. Here, again, the SQP model shows a better fit for the new data compared to the Q-N model.

5. Conclusions

A hybrid algorithm is presented that combines the exploration abilities of GA and the exploitation of CSA. The incorporation of GA resulted in improved CSA convergence and a reduced risk of falling into a local extremum. The hybrid GA-CSA was compared to the pure GA and pure CSA to evaluate its performance. The algorithms were applied for the modeling of the *E. coli BL21(DE3)pPhyt109* fed-batch cultivation process. Parameter identification procedures of a nonlinear model of the considered CP were performed. Based on a set of experimental data, mathematical models were obtained and then successfully verified by a second set of experimental data for the same CP. The best performance was achieved by the proposed hybrid GA-CSA. The results were confirmed by the InterCriteria analysis and statistical tests, both parametric and nonparametric. The analysis indicated that the three metaheuristic algorithms, GA, CSA, and GA-CSA, achieved statistically distinguishable numerical results. The estimated strong dissonance between the pairs GA vs. CSA and CSA vs. GA-CSA, and the weak dissonance between GA vs. GA-CSA, showed that an algorithm with outstanding performance was developed based on the proposed collaborative hybridization. The results obtained by applying metaheuristic algorithms were compared to the results from two deterministic algorithms—the SQP and Q-N methods. The model estimated by SQP showed better quality compared to the Q-N model. Both methods did not perform well compared to GA, CSA, and the hybrid GA-CSA; moreover, they were highly dependent on the given initial solutions.

The proposed hybrid algorithm produced the most accurate model and best described the dynamics of the considered CP of *E. coli BL21(DE3)pPhyt109*. Upon verifying the models, again, the GA-CSA model showed the closest proximity to the available experimental data.

Despite the obvious advantages of the hybrid metaheuristic, it also showed some disadvantages. The resulting parameter estimates of the models had the largest SD, i.e., they varied widely. In the future, efforts will be focused on improving the performance of the GA-CSA method, especially in reducing the high variability of the estimates.

A possible solution to overcome this disadvantage is to use an adaptive variation of the flight length and awareness probability. Applying adaptive algorithm parameters will enhance the balance between exploitation and exploration performance [32–34].

Author Contributions: Conceptualization, O.R. and D.Z.; methodology, O.R.; software, D.Z.; validation, O.R. and D.Z.; formal analysis, O.R. and D.Z.; investigation, O.R. and D.Z.; writing—original draft preparation, O.R. and D.Z.; writing—review and editing, O.R. and D.Z.; visualization, O.R. and D.Z.; supervision, O.R. All authors have read and agreed to the published version of the manuscript.

Funding: This research received funding under Grant Ref. No. KN-06-N-22/1, "Theoretical research and applications of InterCriteria Analysis", funded by the Bulgarian National Science Fund.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data are contained within the article.

Acknowledgments: D.Z. is grateful for Grant KII-06-H32/3, "Interactive System for Education in Modelling and Control of Bioprocesses (InSEMCoBio)", National Scientific Fund of Bulgaria.

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

References

1. Hassanien, A.E.; Rizk-Allah, R.M.; Elhoseny, M. A hybrid crow search algorithm based on rough searching scheme for solving engineering optimization problems. *J. Ambient Intell. Humaniz. Comput.* **2018**, 1–25. [CrossRef]
2. Yang, X.S. Nature-inspired optimization algorithms: Challenges and open problems. *J. Comput. Sci.* **2020**, *46*, 101104. [CrossRef]
3. Hussain, K.; Mohd Salleh, M.N.; Cheng, S.; Shi, Y. Metaheuristic research: A comprehensive survey. *Artif. Intell. Rev.* **2019**, *52*, 2191–2233. [CrossRef]
4. Hu, J.; Wu, H.; Zhong, B.; Xiao, R. Swarm intelligence-based optimisation algorithms: An overview and future research issues. *Int. J. Autom. Control* **2020**, *14*, 656–693. [CrossRef]
5. Sharma, P.; Raju, S. Metaheuristic optimization algorithms: A comprehensive overview and classification of benchmark test functions. *Soft Comput.* **2023**, 1–64. [CrossRef]
6. Pan, J.S.; Hu, P.; Snášel, V.; Chu, S.C. A survey on binary metaheuristic algorithms and their engineering applications. *Artif. Intell. Rev.* **2023**, *56*, 6101–6167. [CrossRef] [PubMed]
7. Kaur, S.; Kumar, Y.; Koul, A.; Kumar Kamboj, S. A systematic review on metaheuristic optimization techniques for feature selections in disease diagnosis: Open issues and challenges. *Arch. Comput. Methods Eng.* **2023**, *30*, 1863–1895. [CrossRef]
8. Ab Wahab, M.N.; Nefti-Meziani, S.; Atyabi, A. A comparative review on mobile robot path planning: Classical or meta-heuristic methods? *Annu. Rev. Control* **2020**, *50*, 233–252. [CrossRef]
9. Nassef, A.M.; Abdelkareem, M.A.; Maghrabie, H.M.; Baroutaji, A. Review of Metaheuristic Optimization Algorithms for Power Systems Problems. *Sustainability* **2023**, *15*, 9434. [CrossRef]
10. Roeva, O.; Zoteva, D.; Castillo, O. Joint set-up of parameters in genetic algorithms and the artificial bee colony algorithm: An approach for cultivation process modelling. *Soft Comput.* **2021**, *25*, 2015–2038. [CrossRef]
11. Roeva, O.; Zoteva, D.; Roeva, G.; Lyubenova, V. An Efficient Hybrid of an Ant Lion Optimizer and Genetic Algorithm for a Model Parameter Identification Problem. *Mathematics* **2023**, *11*, 1292. [CrossRef]
12. Goldberg, D. *Genetic Algorithms in Search, Optimization and Machine Learning*, 1st ed.; Addison-Wesley Professional: Boston, MA, USA, 1989.
13. Yang, X.-S.; Deb, S. Cuckoo search via levy flights. In Proceedings of the World Congress on Nature and Biologically Inspired Computing, Coimbatore, India, 9–11 December 2009; IEEE: Piscataway, NJ, USA, 2009; pp. 210–214.
14. Dorigo, M.; Stutzle, T. *Ant Colony Optimization*; MIT Press: Cambridge, UK, 2004.
15. Askarzadeh, A. A novel metaheuristic method for solving constrained engineering optimization problems: Crow search algorithm. *Comput. Struct.* **2016**, *169*, 1–12. [CrossRef]
16. Angelova, M.; Roeva, O.; Vassilev, P.; Pencheva, T. Multi-Population Genetic Algorithm and Cuckoo Search Hybrid Technique for Parameter Identification of Fermentation Process Models. *Processes* **2023**, *11*, 427. [CrossRef]
17. Angelova, M.; Vassilev, P.; Pencheva, T. Genetic Algorithm and Cuckoo Search Hybrid Technique for Parameter Identification of Fermentation Process Model. *Int. J. Bioaut.* **2020**, *24*, 277–288. [CrossRef]

18. Sun, Y.; Dai, H.L.; Moayedi, H.; Le, B.N.; Adnan, R.M. Predicting steady-state biogas production from waste using advanced machine learning-metaheuristic approaches. *Fuel* **2024**, *355*, 129493. [CrossRef]
19. Zhao, S.; Jiao, T.; Wang, Z.; Adade, S.Y.S.S.; Wu, X.; Ouyang, Q.; Chen, Q. On-line detecting soluble sugar, total acids, and bacterial concentration during kombucha fermentation based on the visible/near infrared combined meta-heuristic algorithm. *J. Food Compos. Anal.* **2023**, *123*, 105653. [CrossRef]
20. Roeva, O.; Angelova, M.; Zoteva, D.; Pencheva, T. Water cycle algorithm for modelling of fermentation processes. *Processes* **2020**, *8*, 920. [CrossRef]
21. Bouaouda, A.; Sayouti, Y. Hybrid meta-heuristic algorithms for optimal sizing of hybrid renewable energy system: A review of the state-of-the-art. *Arch. Comput. Methods Eng.* **2022**, *29*, 4049–4083. [CrossRef]
22. Hassan, A.; Pillay, N. Hybrid metaheuristics: An automated approach. *Expert Syst. Appl.* **2019**, *130*, 132–144. [CrossRef]
23. Zhao, S.; Wang, P.; Heidari, A.A.; Zhao, X.; Chen, H. Boosted crow search algorithm for handling multi-threshold image problems with application to X-ray images of COVID-19. *Expert Syst. Appl.* **2023**, *213*, 119095. [CrossRef]
24. Singh, H.; Tyagi, S.; Kumar, P. Cloud resource mapping through crow search inspired metaheuristic load balancing technique. *Comput. Electr. Eng.* **2021**, *93*, 107221. [CrossRef]
25. Mohammadi, F.; Abdi, H. A modified crow search algorithm (MCSA) for solving economic load dispatch problem. *Appl. Soft Comput.* **2018**, *71*, 51–65. [CrossRef]
26. Meraihi, Y.; Gabis, A.B.; Ramdane-Cherif, A.; Acheli, D. A comprehensive survey of Crow Search Algorithm and its applications. *Artif. Intell. Rev.* **2021**, *54*, 2669–2716. [CrossRef]
27. Wang, T.; Zhang, C.; He, A.; Dong, W. Overview of Crow Search Algorithm. *J. Phys. Conf. Ser.* **2022**, *2258*, 012017. [CrossRef]
28. Hussien, A.G.; Amin, M.; Wang, M.; Liang, G.; Alsanad, A.; Gumaei, A.; Chen, H. Crow search algorithm: Theory, recent advances, and applications. *IEEE Access* **2020**, *8*, 173548–173565. [CrossRef]
29. Huang, K.W.; Girsang, A.S.; Wu, Z.X.; Chuang, Y.W. A hybrid crow search algorithm for solving permutation flow shop scheduling problems. *Appl. Sci.* **2019**, *9*, 1353. [CrossRef]
30. Cheng, Q.; Huang, H.; Chen, M.A. Novel Crow Search Algorithm Based on Improved Flower Pollination. *Hindawi Math. Probl. Eng.* **2021**, *2021*, 1048879. [CrossRef]
31. Sannasi Chakravarthy, S.R.; Rajaguru, H. A novel improved crow-search algorithm to classify the severity in digital mammograms. *Int. J. Imaging Syst. Technol.* **2021**, *31*, 921–954. [CrossRef]
32. Díaz, P.; Pérez-Cisneros, M.; Cuevas, E.; Avalos, O.; Gálvez, J.; Hinojosa, S.; Zaldivar, D. An improved crow search algorithm applied to energy problems. *Energies* **2018**, *11*, 571. [CrossRef]
33. Lee, D.; Kim, J.; Shon, S.; Lee, S. An Advanced Crow Search Algorithm for Solving Global Optimization Problem. *Appl. Sci.* **2023**, *13*, 6628. [CrossRef]
34. Wijayaningrum, V.N.; Putriwijaya, N.N. An improved crow search algorithm for data clustering. *EMITTER Int. J. Eng. Technol.* **2020**, *8*, 86–101. [CrossRef]
35. Jain, M.; Rani, A.; Singh, V. An improved crow search algorithm for high-dimensional problems. *J. Intell. Fuzzy Syst.* **2017**, *33*, 3597–3614. [CrossRef]
36. Gholami, J.; Mardukhi, F.; Zawbaa, H.M. An improved crow search algorithm for solving numerical optimization functions. *Soft Comput.* **2021**, *25*, 9441–9454. [CrossRef]
37. Cortés-Caicedo, B.; Avellaneda-Gómez, L.S.; Montoya, O.D.; Alvarado-Barrios, L.; Álvarez-Arroyo, C. An improved crow search algorithm applied to the phase swapping problem in asymmetric distribution systems. *Symmetry* **2021**, *13*, 1329. [CrossRef]
38. Huang, H.; Liao, H. An improved crow search algorithm with multi strategy disturbance. In *ITM Web of Conferences*; EDP Sciences: Evry, France, 2022; Volume 47, p. 02007.
39. Sharma, M.; Pathik, B. Crow Search Algorithm with Improved Objective Function for Test Case Generation and Optimization. *Intell. Autom. Soft Comput.* **2022**, *32*, 1125–1140. [CrossRef]
40. Shekhawat, S.; Saxena, A. Development and applications of an intelligent crow search algorithm based on opposition based learning. *ISA Trans.* **2020**, *99*, 210–230. [CrossRef]
41. He, J.; Peng, Z.; Zhang, L.; Zuo, L.; Cui, D.; Li, Q. Enhanced crow search algorithm with multi-stage search integration for global optimization problems. *Soft Comput.* **2023**, *27*, 14877–14907. [CrossRef]
42. Javidi, A.; Salajegheh, E.; Salajegheh, J. Enhanced crow search algorithm for optimum design of structures. *Appl. Soft Comput.* **2019**, *77*, 274–289. [CrossRef]
43. Sayed, G.I.; Hassanien, A.E.; Azar, A.T. Feature selection via a novel chaotic crow search algorithm. *Neural Comput. Appl.* **2019**, *31*, 171–188. [CrossRef]
44. Arora, S.; Singh, H.; Sharma, M.; Sharma, S.; Anand, P. A new hybrid algorithm based on grey wolf optimization and crow search algorithm for unconstrained function optimization and feature selection. *IEEE Access* **2019**, *7*, 26343–26361. [CrossRef]
45. El-Ashmawi, W.H.; Ali, A.F.; Slowik, A. Hybrid crow search and uniform crossover algorithm-based clustering for top-N recommendation system. *Neural Comput. Appl.* **2021**, *33*, 7145–7164. [CrossRef]
46. Anter, A.M.; Ali, M. Feature selection strategy based on hybrid crow search optimization algorithm integrated with chaos theory and fuzzy c-means algorithm for medical diagnosis problems. *Soft Comput.* **2020**, *24*, 1565–1584. [CrossRef]
47. Krishnadoss, P.; Pradeep, N.; Ali, J.; Nanjappan, M.; Krishnamoorthy, P.; Kedalu Poornachary, V. CCSA: Hybrid cuckoo crow search algorithm for task scheduling in cloud computing. *Int. J. Intell. Eng. Syst.* **2021**, *14*, 241–250. [CrossRef]

48. Hossain, S.Z.; Sultana, N.; Mohammed, M.E.; Razzak, S.A.; Hossain, M.M. Hybrid support vector regression and crow search algorithm for modeling and multiobjective optimization of microalgae-based wastewater treatment. *J. Environ. Manag.* **2022**, *301*, 113783. [CrossRef] [PubMed]
49. Kumar, S.N.; Fred, A.L.; Miriam, L.J.; Padmanabhan, P.; Gulyás, B.; Kumar, A.; Dayana, N. Improved crow search algorithm based on arithmetic crossover—A novel metaheuristic technique for solving engineering optimization problems. In *Multi-Objective Combinatorial Optimization Problems and Solution Methods*; Academic Press: Cambridge, MA, USA, 2022; pp. 71–91.
50. Cao, L.; Yue, Y.; Zhang, Y.; Cai, Y. Improved crow search algorithm optimized extreme learning machine based on classification algorithm and application. *IEEE Access* **2021**, *9*, 20051–20066. [CrossRef]
51. Gupta, D.; Sundaram, S.; Khanna, A.; Hassanien, A.E.; De Albuquerque, V.H.C. Improved diagnosis of Parkinson's disease using optimized crow search algorithm. *Comput. Electr. Eng.* **2018**, *68*, 412–424. [CrossRef]
52. Hamed Alnaish, Z.A.; Algamal, Z.Y. Improving binary crow search algorithm for feature selection. *J. Intell. Syst.* **2023**, *32*, 20220228. [CrossRef]
53. Braik, M.; Al-Zoubi, H.; Ryalat, M.; Sheta, A.; Alzubi, O. Memory based hybrid crow search algorithm for solving numerical and constrained global optimization problems. *Artif. Intell. Rev.* **2023**, *56*, 27–99. [CrossRef]
54. Durgut, R.; Aydin, M.E. Multi Strategy Search with Crow Search Algorithm. In *Optimisation Algorithms and Swarm Intelligence*; IntechOpen: Rijeka, Croatia, 2022.
55. Adamu, A.; Abdullahi, M.; Junaidu, S.B.; Hassan, I.H. A hybrid particle swarm optimization with crow search algorithm for feature selection. *Mach. Learn. Appl.* **2021**, *6*, 100108. [CrossRef]
56. Roeva, O.; Pencheva, T.; Tzonkov, S.; Arndt, M.; Hitzmann, B.; Kleist, S.; Miksch, G.; Friehs, K.; Flaschel, E. Multiple model approach to modelling of *Escherichia coli* fed-batch cultivation extracellular production of bacterial phytase. *Electron. J. Biotechnol.* **2007**, *10*, 592–603. [CrossRef]
57. Kleist, S.; Miksch, G.; Hitzmann, B.; Arndt, M.; Freihs, K.; Flaschel, E. Optimization of the extracellular production of a bacterial phytase with *Escherichia coli* by using different fed-batch fermentation strategies. *Appl. Microbiol. Biotechnol.* **2003**, *61*, 456–462. [CrossRef] [PubMed]
58. Miksch, G.; Kleist, S.; Friehs, K.; Flaschel, E. Overexpression of the phytase from *Escherichia coli* and its extracellular production in bioreactors. *Appl. Microbiol. Biotechnol.* **2002**, *59*, 685–694. [CrossRef]
59. Atanassov, K.; Mavrov, D.; Atanassova, V. Intercriteria decision making: A new approach for multicriteria decision making, based on index matrices and intuitionistic fuzzy sets. *Issues Intuit. Fuzzy Sets Gen. Nets* **2014**, *11*, 1–8.
60. Zaharieva, B.; Doukovska, L.; Ribagin, S.; Radeva, I. InterCriteria Analysis of Data Obtained from Patients with Behterev's Disease. *Int. J. Bioautom.* **2020**, *24*, 5–14. [CrossRef]
61. Jekova, I.; Vassilev, P.; Stoyanov, T.; Pencheva, T. InterCriteria analysis: Application for ECG data analysis. *Mathematics* **2021**, *9*, 854. [CrossRef]
62. Atanassov, K.; Pencheva, T. InterCriteria Analysis Approach as a Tool for Promising Decision Making in Physiological Rhythms. In *2019-20 MATRIX Annals*; MATRIX Book Series; de Gier, J., Praeger, C.E., Tao, T., Eds.; Springer: Cham, Switzerland, 2021; Volume 4. [CrossRef]
63. Petrov, M. Modeling and Use of Inter-Criteria Decision Analysis for Selecting Growth Rate Models for Batch Cultivation of Yeast Kluyveromyces marxianus var. lactis MC 5. *Fermentation* **2021**, *7*, 163. [CrossRef]
64. Antonov, A. Analysis and Detection of the Degrees and Direction of Correlations between Key Indicators of Physical Fitness of 10–12-year-old Hockey Players. *Int. J. Bioautom.* **2019**, *23*, 303–314. [CrossRef]
65. Fisher, R.A. *Statistical Methods and Scientific Inference*, 2nd ed.; Hafner Publishing Co.: New York, NY, USA, 1959.
66. García, S.; Molina, D.; Lozano, M.; Herrera, F. A study on the use of nonparametric tests for analyzing the evolutionary algorithms' behaviour: A case study on the CEC'2005 special session on real parameter optimization. *J. Heuristics* **2009**, *15*, 617–644. [CrossRef]
67. García, S.; Fernández, A.; Luengo, J.; Herrera, F. Advanced nonparametric tests for multiple comparisons in the design of experiments in computational intelligence and data mining: Experimental analysis of power. *Inf. Sci.* **2010**, *180*, 2044–2064. [CrossRef]
68. Oliva, D.; Hinojosa, S.; Cuevas, E.; Pajares, G.; Avalos, O.; Gálvez, J. Cross entropy based thresholding for magnetic resonance brain images using Crow Search Algorithm. *Expert Syst. Appl.* **2017**, *79*, 164–180. [CrossRef]
69. Velasco, L.; Guerrero, H.; Hospitaler, A. Can the global optimum of a combinatorial optimization problem be reliably estimated through extreme value theory? *Swarm Evol. Comput.* **2022**, *75*, 10117. [CrossRef]
70. Pandey, H.M. 3—State of the Art: Genetic Algorithms and Premature Convergence. In *State of the Art on Grammatical Inference Using Evolutionary Method*; Hari, M.P., Ed.; Academic Press: Cambridge, MA, USA, 2022; pp. 35–124.
71. Roeva, O.; Zoteva, D. Knowledge discovery from data: InterCriteria analysis of mutation rate influence. *Notes Intuitionistic Fuzzy Sets* **2018**, *24*, 120–130. [CrossRef]
72. Roeva, O.; Fidanova, S.; Atanassova, V. Hybrid ACO-GA for parameter identification of an *E. coli* cultivation process model. In *International Conference on Large-Scale Scientific Computing*; Springer: Berlin/Heidelberg, Germany, 2013; pp. 313–320. Available online: https://link.springer.com/chapter/10.1007/978-3-662-43880-0_35 (accessed on 7 November 2023).
73. Mühlenbein, H.; Schlierkamp-Voosen, D. Predictive models for the breeder genetic algorithm I. Continuous parameter optimization. *Evol. Comput.* **1993**, *1*, 25–49. [CrossRef]
74. Atanassov, K.; Szmidt, E.; Kacprzyk, J. On intuitionistic fuzzy pairs. *Notes Intuitionistic Fuzzy Sets* **2013**, *19*, 1–13.

75. Atanassov, K. Intuitionistic Fuzzy Sets. VII ITKR Session, Sofia, 20–23 June 1983. Reprinted. *Int. J. Bioautom.* **2016**, *20*, S1–S6.
76. Atanassov, K.; Atanassova, V.; Gluhchev, G. InterCriteria Analysis: Ideas and problems. *Notes Intuitionistic Fuzzy Sets* **2015**, *21*, 81–88.
77. Ikonomov, N.; Vassilev, P.; Roeva, O. ICrAData—Software for InterCriteria Analysis. *Int. J. Bioautom.* **2018**, *2*, 1–10. [CrossRef]
78. Atanassov, K. On index matrices, Part 1: Standard cases. *Adv. Stud. Contemp. Math.* **2010**, *20*, 291–302.