Journal of Materials Science and Chemical Engineering, 2022, 10, 1-9
https://www.scirp.org/journal/msce
ISSN Online: 2327-6053
ISSN Print: 2327-6045

Investigation into the Computational Costs of Using Genetic Algorithm and Simulated Annealing for the Optimization of Explicit Friction Factor Models

Sunday Boladale Alabi*, Abasiyake Uku Ekpenyong

Department of Chemical and Petroleum Engineering, University of Uyo, Uyo, Nigeria
Email: *sundayalabi@uniuyo.edu.ng

How to cite this paper: Alabi, S. B., & Ekpenyong, A. U. (2022) Investigation into the Computational Costs of Using Genetic Algorithm and Simulated Annealing for the Optimization of Explicit Friction Factor Models. *Journal of Materials Science and Chemical Engineering*, **10**, 1-9.
https://doi.org/10.4236/msce.2022.1012001

Received: December 14, 2022
Accepted: December 27, 2022
Published: December 30, 2022

Copyright © 2022 by author(s) and Scientific Research Publishing Inc.
This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).
http://creativecommons.org/licenses/by/4.0/
Open Access

Abstract

Research reports show that the accuracies of many explicit friction factor models, having different levels of accuracies and complexities, have been improved using genetic algorithm (GA), a global optimization approach. However, the computational cost associated with the use of GA has yet to be discussed. In this study, the parameters of sixteen explicit models for the estimation of friction factor in the turbulent flow regime were optimized using two popular global search methods namely genetic algorithm (GA) and simulated annealing (SA). Based on 1000 interval values of Reynolds number (*Re*) in the range of $4\times10^3 \le Re \le 1\times10^8$ and 100 interval values of relative roughness ($\mathcal{E}/D$) in the range of $10^{-6} \le \mathcal{E}/D \le 5\times10^{-2}$, corresponding friction factor (*f*) data were obtained by solving Colebrook-White equation using Microsoft Excel spreadsheet. These data were then used to modify the parameters of the selected explicit models. Although both GA and SA led to either moderate or significant improvements in the accuracies of the existing friction factor models, SA outperforms the GA. Moreover, the SA requires far less computational time than the GA to complete the corresponding optimization process. It can therefore be concluded that SA is a better global optimizer than GA in the process of finding an improved explicit friction factor model as an alternative to the implicit Colebrook-White equation in the turbulent flow regime.

Keywords

Genetic Algorithm, Simulated Annealing, Global Optimization, Explicit Friction Factor, Computational Cost

1. Introduction

The Colebrook-White model, given in Equation (1), has been widely accepted and used in engineering practices as sufficiently accurate for the calculation of pipe friction factor (f) in the turbulent flow regime [1] [2].

$$\frac{1}{\sqrt{f}} = -2\log\left(\frac{\mathcal{E}/D}{3.71} + \frac{2.51}{Re\sqrt{f}}\right) \tag{1}$$

However, due to the implicit nature of Colebrook-White equation, it requires an iterative solution method such as the Newton-Raphson where numerous calculations are required for long pipelines and network of pipes to obtain the value of f as a function of Reynolds Number (Re) and relative roughness of the pipe ($\mathcal{E}/D$). This method is time-consuming and complicated. Moody provided a diagram called Moody chart [3], as a graphical solution of Colebrook equation. Although the chart eliminates the requirement for iteration, obtaining data from this chart and interpolating the friction factor values is error-prone. Moreover, it is inconvenient for computer simulation. In order to overcome these drawbacks, various explicit friction factor models have been developed using diverse parameter estimation methods.

In comparison with the iterative solution of Colebrook-White equation, the explicit models differ in their accuracies, complexities and relative computational efficiencies (see [4] [5] [6]). Most of the existing explicit friction factor models were developed using local optimization methods. Cojbasić and Brkić [7] selected two models proposed by Serghides [8] and, Romeo and Co-workers [9] and have been proven to be among the most accurate explicit models by Winning and Cooles [10] to modify their parameters using genetic algorithm (GA), a global optimization technique. After successful optimization, the accuracies of these models were improved 53 and 16 times, respectively. Thus, Brkić and Ćojbašić [11] made efforts to improve the accuracies of several explicit models by modifying their parameters using genetic algorithm (GA). Although GA has been used to improve the accuracies of several existing explicit friction factor models, the computational burden associated with its use has not been discussed. Souza and Co-workers [12] used simulated annealing, another global optimization algorithm to directly obtain the parameters of very few explicit friction factor models. Unfortunately, the authors were silent about the computational burdens associated with the use of SA and its use for explicit friction factor modelling and optimization has yet to be widely studied.

Consequently, using the model accuracy and the computational time of the search algorithm, as the performance indicators, this study investigates extensively the effectiveness of the GA and SA for the parameters estimation of several explicit friction factor models.

2. Theoretical Background

2.1. Genetic Algorithm

Genetic algorithms are one of the evolutionary computational intelligence techniques, inspired by Darwin's theory of biological evolution. Genetic algorithms are very powerful tools for optimization. It is a method for solving both constrained and unconstrained optimization problems that is based on natural selection, the process that drives biological evolution. The genetic algorithm repeatedly modifies a population of individual solutions. At each step, the genetic algorithm selects individuals at random from the current population to be parents and uses them to produce the children for the next generation. Over successive generations, the population "evolves" toward an optimal solution. Genetic algorithm (GA) can be applied to solve a variety of optimization problems that are not well suited for standard optimization algorithms, including problems in which the objective function is discontinuous, non-differentiable, stochastic, or highly nonlinear. The genetic algorithm can address problems of mixed integer programming, where some components are restricted to be integer-valued. The fundamental details about this algorithm are beyond the scope of this study; interested Reader is referred to MathWorks' Global Optimization Toolbox User Guide [13].

In genetic algorithm, the function that needs to be optimized is called fitness function. The fitness function which is the explicit model is coded and passed as a function handle input argument to the main genetic algorithm function. While coding the initial numerical coefficient is specified, "x". These are the numbers of variables in the fitness function. These x components are provided with lower and upper bounds for GA to search for the best optimal solution. These upper and lower bounds are within the neighborhood of the initial values.

2.2. Simulated Annealing

Simulated annealing (SA) is a method for solving unconstrained and bound-constrained optimization problems. SA is one of the most flexible techniques available for solving hard combinatorial problems. The main advantage of SA is that it can be applied to large problems regardless of the conditions of differentiability, continuity, and convexity than are normally required in conventional methods. Annealing is a heat treatment process where a material is subjected to high temperature, with subsequent cooling, so as to obtain high-quality crystals (*i.e.*, crystals whose structure form perfect lattices). During the cooling process, it is assumed that thermal equilibrium (or quasi equilibrium) conditions are maintained. The cooling process ends when the material reaches a state of minimum energy, which, in principle, corresponds with a perfect crystal. It is known that defect-free crystals (*i.e.*, solids with minimum energy) are more likely to be formed under a slow cooling process. The two main features of the simulated annealing process are:

1) The transition mechanism between states and

2) The cooling schedule.

When applied to combinatorial optimization, simulated annealing aims to find an optimal configuration (or state with minimum "energy") of a complex problem. The objective function of an optimization problem corresponds with the free energy of the material. An optimal solution is associated with a perfect crystal, whereas a crystal with defects corresponds with a local optimal solution. The analogy is not complete, however, because in the annealing process there is a physical variable that is the temperature, which under proper control leads to the formation of a perfect crystal.

When simulated annealing is used as an optimization technique, the temperature becomes simply a control parameter that has to be properly determined in order to achieve the desired results. The temperature affects two aspects of the algorithm: the distance of a trial point from the current point and the probability of accepting a trial point with higher objective function value.

Temperature can be a vector with different value for each component of the current point. The initial temperature is a scalar. Temperature decreases gradually as the algorithm proceeds. The slower the rate of temperature decreases, the better the chances of finding the optimal solution. The fundamental details about this algorithm are beyond the scope of this study; interested Reader is referred to MathWorks' Global Optimization Toolbox User Guide [13].

3. Materials and Method

3.1. Data Generation

Sixteen explicit models with excellent performance and high precision which can be used as alternative to Colebrook white equation were selected from the literature and analyzed. The friction factor (f) of the equations were generated using Microsoft Excel spreadsheet with 1000 interval values of Reynolds number (Re) in the range of $4\times10^3 \le Re \le 1\times10^8$ and 100 interval values of relative roughness ($\mathcal{E}/D$) in the range of $10^{-6} \le \mathcal{E}/D \le 5\times10^{-2}$. Subsequently, the Re and $\mathcal{E}/D$ data were used to iteratively solve Colebrook-White equation (Equation (1)).

3.2. Error Estimation

The measure of deviation of each explicit model prediction from the iterative solution of Colebrook-White equation can be based on absolute error, mean square error (MSE), average error and absolute relative error (%). In this paper, we focused on the maximum absolute relative error (MARE) of each explicit model to estimate their level of accuracy. The data obtained in Section 3.1 were used to measure each deviation. Absolute relative error is computed using the formula given in Equation (2)

$$\text{Absolute Relative Error}(\%) = \frac{\left| f_{\text{colebrook}} - f_{\text{explicit}} \right|}{f_{\text{colebrook}}} \times 100 \qquad (2)$$

3.3. Optimization

3.3.1. Genetic Algorithm Optimization

The optimization with the genetic algorithm was carried out using GA tool in the Global optimization toolbox in MATLAB installed HP Intel® Pentium® CPU

N3540 @ 2.16 GHz Windows 10 Laptop Computer. The data obtained in section 3.1 were used in the optimization process to obtain new parameters for each explicit model. The models before and after optimization with GA as well as the time for the completion of each optimization process are presented in Table 1 and Table 2, respectively.

Table 1. Explicit models to Colebrook-White equation before and after optimization with genetic algorithm (GA) and simulated annealing (SA).

Model/Reference	Model before optimization	Model after optimization with GA	Model after optimization with SA
Chen [14]	$\frac{1}{\sqrt{f}}=-2\log\left[\frac{\mathcal{E}}{3.7065D}-\frac{5.0452}{Re}\log_{10}\left[\frac{1}{2.8257}\cdot\left(\frac{\mathcal{E}}{D}\right)^{1.1098}+\frac{5.8506}{Re^{0.8981}}\right]\right]$	$\frac{1}{\sqrt{f}}=-2.002\log\left[\frac{\mathcal{E}}{3.7D}-\frac{4.975}{Re}\log_{10}\left[\frac{1}{3.185}\cdot\left(\frac{\mathcal{E}}{D}\right)^{1.078}+\frac{5.495}{Re^{0.908}}\right]\right]$	$\frac{1}{\sqrt{f}}=-2.002\log\left[\frac{\mathcal{E}}{3.707D}-\frac{5.262}{Re}\log_{10}\left[\frac{1}{3.111}\cdot\left(\frac{\mathcal{E}}{D}\right)^{1.02}+\frac{5.504}{Re^{0.869}}\right]\right]$
Barr [15]	$\frac{1}{\sqrt{f}}=-2\log_{10}\left[\frac{\mathcal{E}}{3.7D}+\frac{4.518\log\left(\frac{1}{7}Re\right)}{Re\left(1+\frac{1}{29}Re^{0.52}\left(\frac{\mathcal{E}}{D}\right)^{0.7}\right)}\right]$	$\frac{1}{\sqrt{f}}=-2.001\log_{10}\left[\frac{\mathcal{E}}{3.69D}+\frac{4.571\log\left(\frac{1}{7.114}Re\right)}{Re\left(1+\frac{1}{30.407}Re^{0.605}\left(\frac{\mathcal{E}}{D}\right)^{0.842}\right)}\right]$	$\frac{1}{\sqrt{f}}=-2.001\log_{10}\left[\frac{\mathcal{E}}{3.697D}+\frac{4.56\log\left(\frac{1}{7.013}Re\right)}{Re\left(1+\frac{1}{29.224}Re^{0.586}\left(\frac{\mathcal{E}}{D}\right)^{0.821}\right)}\right]$
Papaevangelou *et al.* [16]	$f=\frac{0.2479-0.0000947(7-\log Re)^{4}}{\left[\log\left[\frac{\mathcal{E}}{3.615D}+\frac{7.366}{Re^{0.9142}}\right]\right]^{2}}$	$f=\frac{0.246-0.0000982(7.192-\log Re)^{3.677}}{\left[\log\left[\frac{\mathcal{E}}{3.556D}+\frac{6.756}{Re^{0.905}}\right]\right]^{2}}$	$f=\frac{0.247-0.0000982(7.008-\log Re)^{3.889}}{\left[\log\left[\frac{\mathcal{E}}{3.593D}+\frac{7.174}{Re^{0.912}}\right]\right]^{2}}$
Avci and Karagoz [17]	$f=\frac{6.4}{\left[In(Re)-In\left(1+0.01Re\frac{\mathcal{E}}{D}\left(1+10\sqrt{\frac{\mathcal{E}}{D}}\right)\right)\right]^{2.4}}$	$f=\frac{6.609}{\left[In(Re)-In\left(1.082+0.009Re\frac{\mathcal{E}}{D}\left(1.286+9.896\sqrt{\frac{\mathcal{E}}{D}}\right)\right)\right]^{2.418}}$	$f=\frac{6.831}{\left[In(Re)-In\left(1.076+0.01Re\frac{\mathcal{E}}{D}\left(1.289+8.098\sqrt{\frac{\mathcal{E}}{D}}\right)\right)\right]^{2.434}}$
Offor and Alabi [4]	$f=\left[-2\log_{10}\left[\frac{\mathcal{E}/D}{3.71}+\frac{-1.975}{Re}\left[In\left[\left(\frac{\mathcal{E}/D}{3.93}\right)^{1.092}+\frac{7.627}{Re+395.9}\right]\right]\right]\right]^{-2}$	$f=\left[-2\log_{10}\left[\frac{\mathcal{E}/D}{3.712}+\frac{-1.972}{Re}\left[In\left[\left(\frac{\mathcal{E}/D}{3.992}\right)^{1.086}+\frac{7.541}{Re+402}\right]\right]\right]\right]^{-2}$	$f=\left[-2\log_{10}\left[\frac{\mathcal{E}/D}{3.707}+\frac{-1.972}{Re}\left[In\left[\left(\frac{\mathcal{E}/D}{3.861}\right)^{1.088}+\frac{7.533}{Re+397.41}\right]\right]\right]\right]^{-2}$
Shacham [18]	$\frac{1}{\sqrt{f}}=-2\log\left[\frac{\mathcal{E}}{3.7D}-\frac{5.02}{Re}\log\left(\frac{\mathcal{E}}{3.7D}+\frac{14.5}{Re}\right)\right]$	$\frac{1}{\sqrt{f}}=-2.007\log\left[\frac{\mathcal{E}}{3.68D}-\frac{5.133}{Re}\log\left(\frac{\mathcal{E}}{3.71D}+\frac{14.028}{Re}\right)\right]$	$\frac{1}{\sqrt{f}}=-2.008\log\left[\frac{\mathcal{E}}{3.65D}-\frac{5.125}{Re}\log\left(\frac{\mathcal{E}}{3.72D}+\frac{13.5}{Re}\right)\right]$
Sousa *et al.* [12]	$\frac{1}{\sqrt{f}}=-2\log\left[\frac{\mathcal{E}}{3.7D}-\frac{5.16}{Re}\log\left(\frac{\mathcal{E}}{3.7D}+\frac{5.09}{Re^{0.87}}\right)\right]$	$\frac{1}{\sqrt{f}}=-2.001\log\left[\frac{\mathcal{E}}{3.697D}-\frac{5.15}{Re}\log\left(\frac{\mathcal{E}}{3.71D}+\frac{5.06}{Re^{0.873}}\right)\right]$	$\frac{1}{\sqrt{f}}=-2.001\log\left[\frac{\mathcal{E}}{3.7D}-\frac{5.203}{Re}\log\left(\frac{\mathcal{E}}{3.7D}+\frac{5.078}{Re^{0.867}}\right)\right]$
Manadilli [19]	$\frac{1}{\sqrt{f}}=-2\log_{10}\left[\frac{95}{R^{0.983}}-\frac{96.82}{R}+\frac{\varepsilon}{3.7D}\right]$	$\frac{1}{\sqrt{f}}=-2.015\log_{10}\left[\frac{95.01}{R^{0.981}}-\frac{98.623}{R}+\frac{\varepsilon}{3.693D}\right]$	$\frac{1}{\sqrt{f}}=-2.012\log_{10}\left[\frac{94.913}{R^{0.983}}-\frac{96.742}{R}+\frac{\varepsilon}{3.72D}\right]$
Zigrang and Sylvester [20]	$\frac{1}{\sqrt{f}}=-2\log_{10}\left(\frac{\varepsilon}{3.7D}-\frac{5.02}{Re}\cdot\log_{10}\left(\frac{\varepsilon}{3.7D}-\frac{5.02}{Re}\cdot\log_{10}\left(\frac{\varepsilon}{3.7D}-\frac{13}{Re}\right)\right)\right)$	$\frac{1}{\sqrt{f}}=-1.999\log_{10}\left(\frac{\varepsilon}{3.72D}-\frac{5.001}{Re}\cdot\log_{10}\left(\frac{\varepsilon}{3.63D}-\frac{4.91}{Re}\cdot\log_{10}\left(\frac{\varepsilon}{3.9D}-\frac{11.5}{Re}\right)\right)\right)$	$\frac{1}{\sqrt{f}}=-1.999\log_{10}\left(\frac{\varepsilon}{3.719D}-\frac{5.034}{Re}\cdot\log_{10}\left(\frac{\varepsilon}{3.65D}-\frac{5.11}{Re}\cdot\log_{10}\left(\frac{\varepsilon}{3.715D}-\frac{12.5}{Re}\right)\right)\right)$
Ghanbari *et al.* [21]	$f=\left\{-1.52\log\left[\left(\frac{\mathcal{E}}{7.21D}\right)^{1.042}+\left(\frac{2.731}{Re}\right)^{0.9152}\right]\right\}^{-2.169}$	$f=\left\{-1.463\log\left[\left(\frac{\mathcal{E}}{7.568D}\right)^{1.047}+\left(\frac{2.639}{Re}\right)^{0.93}\right]\right\}^{-2.2}$	$f=\left\{-1.482\log\left[\left(\frac{\mathcal{E}}{6.335D}\right)^{1.098}+\left(\frac{2.876}{Re}\right)^{0.96}\right]\right\}^{-2.153}$
Swamme and Jain [22]	$\frac{1}{\sqrt{f}}=-2\log_{10}\left[\frac{\varepsilon}{3.7D}+\frac{5.74}{R^{0.9}}\right]$	$\frac{1}{\sqrt{f}}=-2.009\log_{10}\left[\frac{\varepsilon}{3.78D}+\frac{5.398}{R^{0.895}}\right]$	$\frac{1}{\sqrt{f}}=-2.017\log_{10}\left[\frac{\varepsilon}{3.708D}+\frac{5.219}{R^{0.889}}\right]$
Fang *et al.* [23]	$f=1.613\left[\ln\left(0.237\left(\frac{\varepsilon}{D}\right)^{1.1007}-\frac{60.525}{Re^{1.1105}}+\frac{56.291}{Re^{1.0712}}\right)\right]^{-2}$	$f=1.617\left[\ln\left(0.231\left(\frac{\varepsilon}{D}\right)^{1.1004}-\frac{60.5}{Re^{1.1103}}+\frac{56.69}{Re^{1.072}}\right)\right]^{-2}$	$f=1.603\left[\ln\left(0.231\left(\frac{\varepsilon}{D}\right)^{1.095}-\frac{60.355}{Re^{1.108}}+\frac{56.14}{Re^{1.069}}\right)\right]^{-2}$
Romeo *et al.* [9]	$\frac{1}{\sqrt{f}}=-2\log\left(\frac{\varepsilon/D}{3.7065}-\frac{5.0272}{Re}\log\left(\frac{\varepsilon/D}{3.827}-\frac{4.567}{Re}\cdot\log\left(\left(\frac{\varepsilon/D}{7.7918}\right)^{0.9924}+\left(\frac{5.3326}{208.815+Re}\right)^{0.9345}\right)\right)\right)$	$\frac{1}{\sqrt{f}}=-2\log\left(\frac{\varepsilon/D}{3.7105}-\frac{5}{Re}\log\left(\frac{\varepsilon/D}{3.886}-\frac{4.792}{Re}\cdot\log\left(\left(\frac{\varepsilon/D}{7.74}\right)^{0.969}+\left(\frac{5.01}{212.2+Re}\right)^{0.876}\right)\right)\right)$	$\frac{1}{\sqrt{f}}=-2\log\left(\frac{\varepsilon/D}{3.71}-\frac{5.002}{Re}\log\left(\frac{\varepsilon/D}{3.807}-\frac{4.376}{Re}\cdot\log\left(\left(\frac{\varepsilon/D}{7.858}\right)^{1.008}+\left(\frac{5.422}{206.658+Re}\right)^{0.973}\right)\right)\right)$
Sonnad and Goudar [24]	$\frac{1}{\sqrt{f}}=0.8686\ln\left(\frac{0.4587\cdot Re}{s^{\frac{s}{s+1}}}\right)$ where $s=0.124\cdot Re\cdot\frac{\varepsilon}{D}+\ln(0.4587\cdot Re)$	$\frac{1}{\sqrt{f}}=0.868\ln\left(\frac{0.468\cdot Re}{s^{\frac{s}{s+1}}}\right)$ where $s=0.126\cdot Re\cdot\frac{\varepsilon}{D}+\ln(0.422\cdot Re)$	$\frac{1}{\sqrt{f}}=0.868\ln\left(\frac{0.466\cdot Re}{s^{\frac{s}{s+1}}}\right)$ where $s=0.125\cdot Re\cdot\frac{\varepsilon}{D}+\ln(0.41\cdot Re)$

Continued

Serghides [8]	$f = s_1 - \frac{(s_2 - s_1)^2}{s_3 - 2s_2 + s_1}$ where $s_1 = -2\log_{10}\left(\frac{\varepsilon}{3.7D} + \frac{12}{Re}\right)$ $s_2 = -2\log_{10}\left(\frac{\varepsilon}{3.7D} + \frac{2.51s_1}{Re}\right)$ $s_3 = -2\log_{10}\left(\frac{\varepsilon}{3.7D} + \frac{2.51s_2}{Re}\right)$	$f = s_1 - \frac{(s_2 - s_1)^2}{s_3 - 2s_2 + s_1}$ where $s_1 = -2\log_{10}\left(\frac{\varepsilon}{3.71D} + \frac{12.97}{Re}\right)$ $s_2 = -2\log_{10}\left(\frac{\varepsilon}{3.71D} + \frac{2.51s_1}{Re}\right)$ $s_3 = -2\log_{10}\left(\frac{\varepsilon}{3.71D} + \frac{2.51s_2}{Re}\right)$	$f = s_1 - \frac{(s_2 - s_1)^2}{s_3 - 2s_2 + s_1}$ where $s_1 = -2\log_{10}\left(\frac{\varepsilon}{3.71D} + \frac{11.919}{Re}\right)$ $s_2 = -2\log_{10}\left(\frac{\varepsilon}{3.71D} + \frac{2.51s_1}{Re}\right)$ $s_3 = -2\log_{10}\left(\frac{\varepsilon}{3.71D} + \frac{2.51s_2}{Re}\right)$
Buzzelli [25]	$\frac{1}{\sqrt{f}} = A - \frac{A + 2\log_{10}\left(\frac{B}{Re}\right)}{1 + \frac{2.18}{B}}$ $A = \frac{0.774 In(Re) - 1.41}{1 + 1.32\sqrt{\frac{\varepsilon}{D}}}$ $B = \frac{\varepsilon Re}{3.7D} + 2.51A$	$\frac{1}{\sqrt{f}} = A - \frac{A + 2.002\log_{10}\left(\frac{B}{Re}\right)}{1.001 + \frac{2.033}{B}}$ $A = \frac{0.765 In(Re) - 1.486}{1.139 + 1.491\sqrt{\frac{\varepsilon}{D}}}$ $B = \frac{\varepsilon Re}{3.69D} + 2.522A$	$\frac{1}{\sqrt{f}} = A - \frac{A + 2\log_{10}\left(\frac{B}{Re}\right)}{1 + \frac{2.26}{B}}$ $A = \frac{0.783 In(Re) - 1.383}{0.952 + 1.298\sqrt{\frac{\varepsilon}{D}}}$ $B = \frac{\varepsilon Re}{3.708D} + 2.511A$

Table 2. Performance of the optimized explicit friction factor models.

Models/Reference	Maximum absolute relative error before optimization (%)	Maximum absolute relative error after optimization using GA (%)	Maximum absolute relative error after optimization using SA (%)	Time taken for GA optimization	Time taken for SA optimization
Chen [14]	0.3559	0.1422	0.1315	5 hrs 5 mins	30 mins
Barr [15]	0.5260	0.2113	0.1907	4 hrs 55 mins	45 mins
Papaevangelou *et al.* [16]	0.6974	0.5471	0.4594	9 hrs 35 mins	1 hr 15 mins
Avci and Karagoz [17]	3.0302	1.7975	1.5102	4 hrs 20 mins	35 mins
Offor and Alabi [4]	0.0664	0.0594	0.0583	6 hrs 35 mins	1 hr 20 mins
Shacham [18]	0.8678	0.6153	0.5479	2 hrs 45 mins	25 mins
Sousa *et al.* [12]	0.1658	0.0983	0.095	5 hrs 25 mins	40 mins
Manadilli [19]	2.8232	1.3991	1.3413	1 hr 6 mins	22 mins
Zigrang and Sylvester [20]	0.1255	0.0854	0.0522	5 hrs 12 mins	17 mins
Ghanbari *et al.* [21]	2.7744	1.4083	0.9634	1 hr 45 mins	20 mins
Swamme and Jain [22]	3.4347	1.7477	1.630	2 hrs 26 mins	25 mins
Fang *et al.* [23]	0.5997	0.3919	0.3391	1 hr 26 mins	18 mins
Romeo *et al.* [9]	0.1462	0.0102	0.0032	3 hrs 34 mins	20 mins
Sonnad and Goudar [24]	0.5394	0.0618	0.0479	2 hrs 40 mins	15 mins
Serghides [8]	0.1255	0.0023	0.0017	4 hrs 20 mins	16 mins
Buzzelli [25]	0.1255	0.0669	0.025	12 hrs	46 mins

3.3.2. Simulated Annealing Optimization

Simulated annealing uses temperature parameter to control its global search. The temperature parameter starts off high and is slowly cooled during each iteration step. As it is in GA, the explicit model to be optimized is the objective function. Each explicit model is coded with all numerical variables taking input argument, "*x*". For the minimization of the objective function, it is passed in a function handle to the objective function as well as specifying the start point for each variable. Since it is a bound constraint, the lower and upper bounds for each variable is imputed as a vector. The range of the values is arbitrary and at neighborhood of the initial values. The algorithm searches within the range for the best points that optimize the objective function. Data from section 3.1 were used for the optimization of each model, where optimized parameters were obtained for each input argument, x.

3.4. Performance Evaluation of the Optimized Explicit Models

The deviations of the outputs of the original explicit models and the optimized explicit models from the output of Colebrook-White equation were obtained. The results in terms of maximum absolute relative errors (MAREs) are summarized in Table 2. Moreover, the times taken for the optimized solutions to be obtained for both GA and SA are also presented in Table 2.

4. Discussion

From Table 2, it is observed that the performances of the optimized models using both GA and SA improve either moderately or significantly as they give rise to lower MAREs when compared with those of the original models before optimization. It is noteworthy that MAREs obtained for the SA-optimized models are lower than those for GA-optimized models which indicate that SA is a superior optimizer when compared to GA in terms of accuracy.

Furthermore, from Table 2, it is seen that the times taken by GA to find the optimal solutions for each model are significantly larger than the corresponding times taken by SA. Thus, it is obvious that it is computationally cheaper to use SA for the optimization of the explicit friction models.

Therefore, since SA leads to more accurate explicit friction factor models and requires less computational times than the GA, it can be concluded that SA is a better optimizer than GA in the process of finding an improved explicit friction factor model as an alternative to the implicit Colebrook-White equation.

5. Conclusion and Recommendation

Existing reports show that the performances of the explicit friction factor models developed in lieu of the Colebrook-White equation improve when genetic algorithm (GA), a global optimization method was used to modify their parameters. Unfortunately, the computational costs associated with this process was neither investigated nor reported.

Consequently, this study has successfully investigated the computational burden associated with the use of GA in optimizing the parameters of several existing explicit friction factor models. Moreover, the effectiveness of Simulated Annealing (SA), another global optimization search algorithm was investigated.

Although both GA and SA led to either moderate or significant improvements in the accuracies of the existing friction factor models, SA outperforms the GA.

Moreover, the SA requires far less computational time than the GA to complete the corresponding optimization process.

It can therefore be concluded that SA is a better global optimizer than GA in the process of finding an improved explicit friction factor model as an alternative to the implicit Colebrook-White equation.

The current work is limited to the use of GA and SA. It is therefore recommended that the effectiveness of global optimization methods other than the GA and SA should be investigated for estimating the parameters of the explicit friction factor models.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

[1] Colebrook, C.F. and White, C.M. (1937) Experiments with Fluid Friction Factor in Roughened Pipes. *Proceedings of the Royal Society of London. Series A, Mathematical and Physical Sciences*, **161**, 367-381. https://doi.org/10.1098/rspa.1937.0150

[2] Colebrook, C.F. (1939) Turbulent Flow Pipe Particular Reference to the Transition Region between the Smooth and Rough Pipe Law. *Journal of the Institution of Civil Engineers*, **11**, 133-156. https://doi.org/10.1680/ijoti.1939.13150

[3] Moody, L.F. (1944) Friction Factors for Pipe Flow. *Transactions of the ASME*, **66**, 671-684. https://doi.org/10.1115/1.4018140

[4] Offor, U.H. and Alabi, S.B. (2016) An Accurate and Computationally Efficient Explicit Friction Factor Model. *Advances in Chemical Engineering and Science*, **6**, 237-245. https://doi.org/10.4236/aces.2016.63024

[5] Pérez-Pupo, J., Navarro-Ojeda, M., Pérez-Guerrero, J. and Batista-Zaldívar, M. (2019) On the Explicit Expressions for the Determination of the Friction Factor in Turbulent Regime. *Revista Mexicana de Ingeniería Química*, **19**, 313-334. https://doi.org/10.24275/rmiq/Fen497

[6] Niazkar, M. and Talebbeydokhti, N. (2020) Comparison of Explicit Relations for Calculating Colebrook Friction Factor in Pipe Network Analysis Using h-Based Methods. *Iranian Journal of Science and Technology—Transactions of Civil Engineering*, **44**, 231-249. https://doi.org/10.1007/s40996-019-00343-2

[7] Cojbasić, Ž. and Brkić, D. (2013) Very Accurate Explicit Approximations for Calculation of the Colebrook Friction Factor. *International Journal of Mechanical Sciences*, **67**, 10-13. https://doi.org/10.1016/j.ijmecsci.2012.11.017

[8] Serghides, T.K. (1984) Estimate Friction Factor Accurately. *Chemical Engineering*, **91**, 63-64.

[9] Romeo, E., Royo, C. and Monzón, A. (2002) Improved Explicit Equations for Estimation of the Friction Factor in Rough and Smooth Pipes. *Chemical Engineering Journal*, **86**, 369-374. https://doi.org/10.1016/S1385-8947(01)00254-6

[10] Winning, H.K. and Coole, T. (2013) Explicit Friction Factor Accuracy and Computational Efficiency for Turbulent Flow in Pipes. *Flow, Turbulence and Combustion*, **90**, 1-27. https://doi.org/10.1007/s10494-012-9419-7

[11] Brkić, D. and Ćojbašić, Ž. (2017) Evolutionary Optimization of Colebrook's Turbu-

lent Flow Friction Approximations. *Journal of Fluids Engineering*, **2**, 1-27. https://doi.org/10.20944/preprints201703.0015.v1

[12] Sousa, J., Cunha, M.C. and Marques, A.S. (1999) An Explicit Solution of the Colebrook-White Equation through Simulated Annealing. *Water Industry Systems: Modelling and Optimization Applications*, **2**, 347-355.

[13] MathWorks, Inc. (2015) Global Optimization Toolbox User's Guide. Natick, 604 p.

[14] Chen, N.H. (1979) An Explicit Equation for Friction Factor in Pipe. *Industrial & Engineering Chemistry Research*, **18**, 296-297. https://doi.org/10.1021/i160071a019

[15] Barr, D.I.H. (1981) Solutions of the Colebrook-White Functions for Resistance to Uniform Turbulent Flows. *Proceedings of the Institution of Civil Engineers*, **71**, 529-536. https://doi.org/10.1680/iicep.1981.1895

[16] Papaevangelou, G., Evangelides, C. and Tzimopoulos, C. (2010) A New Explicit Relation for Friction Coefficient in the Darcy-Weisbach Equation. *Proceedings of the 10th Conference on Protection and Restoration of the Environment*, Volume 166, 1-7.

[17] Avci, A. and Karagoz, I. (2009) A Novel Explicit Equation for Friction Factor in Smooth and Rough Pipes. *ASME Journal Fluids Eng.*, **131**, 61-203. https://doi.org/10.1115/1.3129132

[18] Shacham, M. (1980) An Explicit Equation for Friction Factor in Pipe. *Industrial & Engineering Chemistry Fundamentals*, **19**, 228-229. https://doi.org/10.1021/i160074a019

[19] Manadilli, G. (1997) Replace Implicit Equations with Signomial Functions. *Chemical Engineering*, **104**, 129-132.

[20] Zigrang, D.J. and Sylvester, N.D. (1982) Explicit Approximations to the Colebrook's Friction Factor. *AIChE Journal*, **28**, 514-515. https://doi.org/10.1002/aic.690280323

[21] Ghanbari, A., Farshad, F. and Rieke, H.H. (2011) Newly Developed Friction Factor Correlation for Pipe Flow and Flow Assurance. *Journal of Chemical Engineering and Materials Science*, **2**, 83-86.

[22] Swamee, D.K. and Jain, A.K. (1976) Explicit Equations for Pipe Flow Problems. *Journal of the Hydraulics Division*, **102**, 657-664. https://doi.org/10.1061/JYCEAJ.0004542

[23] Fang, X., Xu, Y. and Zhou, Z. (2011) New Correlations of Single-Phase Friction Factor for Turbulent Pipe Flow and Evaluation of Existing Single-Phase Friction Factor Correlations. *Nuclear Engineering and Design*, **241**, 897-902. https://doi.org/10.1016/j.nucengdes.2010.12.019

[24] Sonnad, J.R. and Goudar, C.T. (2006) Turbulent Flow Friction Factor Calculation Using a Mathematically Exact Alternative to the Colebrook-White Equation. *Journal of Hydraulic Engineering*, **132**, 863-867. https://doi.org/10.1061/(ASCE)0733-9429(2006)132:8(863)

[25] Buzzelli, D. (2008) Calculating Friction in One Step. *Machine Design*, **80**, 54-55.