



**HAL**  
open science

# Rational Approximation for Solving an Implicitly Given Colebrook Flow Friction Equation

Pavel Praks, Dejan Brkić

► **To cite this version:**

Pavel Praks, Dejan Brkić. Rational Approximation for Solving an Implicitly Given Colebrook Flow Friction Equation. Mathematics , 2019, 8 (1), pp.26. 10.3390/math8010026 . hal-02438113

**HAL Id: hal-02438113**

**<https://hal.science/hal-02438113>**

Submitted on 14 Jan 2020

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Communication

# Rational Approximation for Solving an Implicitly Given Colebrook Flow Friction Equation

Pavel Praks<sup>1,2,\*</sup>  and Dejan Brkić<sup>1,2,3,\*</sup> <sup>1</sup> European Commission, Joint Research Centre (JRC), 21027 Ispra, Italy<sup>2</sup> IT4Innovations, VŠB—Technical University of Ostrava, 708 00 Ostrava, Czech Republic<sup>3</sup> Research and Development Center “Alfatec”, 18000 Niš, Serbia

\* Correspondence: pavel.praks@vsb.cz (P.P.); dejanrgf@tesla.rcub.bg.ac.rs or dejan.brkic@vsb.cz (D.B.)

Received: 21 November 2019; Accepted: 13 December 2019; Published: 20 December 2019



**Abstract:** The empirical logarithmic Colebrook equation for hydraulic resistance in pipes implicitly considers the unknown flow friction factor. Its explicit approximations, used to avoid iterative computations, should be accurate but also computationally efficient. We present a rational approximate procedure that completely avoids the use of transcendental functions, such as logarithm or non-integer power, which require execution of the additional number of floating-point operations in computer processor units. Instead of these, we use only rational expressions that are executed directly in the processor unit. The rational approximation was found using a combination of a Padé approximant and artificial intelligence (symbolic regression). Numerical experiments in Matlab using 2 million quasi-Monte Carlo samples indicate that the relative error of this new rational approximation does not exceed 0.866%. Moreover, these numerical experiments show that the novel rational approximation is approximately two times faster than the exact solution given by the Wright omega function.

**Keywords:** hydraulic resistance; pipe flow friction; Colebrook equation; Colebrook–White experiment; floating-point computations; approximations; Padé polynomials; symbolic regression

## 1. Introduction

The Colebrook equation [1] for turbulent flow friction is implicitly given with respect to the unknown Darcy flow friction  $\lambda$ , as shown in Equation (1):

$$\frac{1}{\sqrt{\lambda}} = -2 \cdot \log_{10} \left( \frac{2.51}{Re} \cdot \frac{1}{\sqrt{\lambda}} + \frac{\varepsilon}{3.71} \right) \quad (1)$$

where:

$\lambda$ —Darcy flow friction factor (dimensionless)

$Re$ —Reynolds number,  $4000 < Re < 10^8$  (dimensionless)

$\varepsilon$ —relative roughness of inner pipe surface,  $0 < \varepsilon < 0.05$  (dimensionless)

As a PhD student at the Imperial College in London, Colebrook developed his empirical equation based on the data from his joint experiment with his supervisor, Prof. White [2]. They experimented with flow of air through pipes with different roughness of the inner pipe surface. The experiment by Colebrook and White was described in a scientific journal and published in 1937 [2], while the related empirical equation by Colebrook was published in 1939 [1].

Compared with some other experimental findings [3], the Colebrook equation fits the friction factor within a few dozen percent of error [4]. The Colebrook equation over the last 80 years has been seen by the industry as an informal standard for flow friction calculation and has been very

well accepted in everyday engineering practice. Based on the Colebrook equation, Moody developed a diagram that was used before the era of computers for graphical determination of turbulent flow friction [5]. Today, such nomograms have been replaced by explicit approximations, which introduce some value of error [6,7], or by iterative methods [8–10].

In a central computer processor (CPU), transcendental functions such as logarithmic, exponential, or functions with non-integer terms require execution of numerous floating-point operations, and therefore they should be avoided whenever possible [11–18]. Praks and Brkić [19] recently developed a one-log call iterative method, which uses only one computationally demanding function, and even then only in the first iteration (for all succeeding iterations, cheap Padé approximants are used [20,21]). Based on that approach, few very accurate and efficient explicit approximations suitable for coding and for engineering practice have been constructed [22]. In addition, the same authors developed few approximations of the Colebrook equation based on the Wright  $\omega$ -function, which are among the most accurate to date [23–25]. On the other hand, they contain one or two logarithmic functions, depending on the chosen version [23,24] (these procedures are based on the previous efforts by Praks and Brkić for symbolic regression [26] and by Brkić with Lambert W-function [27–29]).

In this communication, we make a step forward and we offer for the first time a procedure for the approximate solution of the Colebrook equation based only on rational functions. The presented novel rational approximation procedure introduces a relative error of no more than 0.866% for  $0 < \varepsilon < 0.05$  and  $4000 < Re < 10^8$  (as used in engineering practice). The rational approximation procedure is suitable for computer codes (open-source code in commercial Matlab 2019a is given in this communication, and in addition, it is compatible with freeware GNU Octave, version 5.1.0).

After introductory Section 1, Section 2 of this communication gives a short overview of mathematical methods used for the proposed rational approximation procedure. Section 3 describes the rational approximation procedure in detail (including error analysis), Section 4 provides software code along with the algorithm to be followed for the rational approximation approach, while Section 5 contains concluding remarks.

## 2. Mathematics Behind the Proposed Approximation

Our rational approximation approach is based on Padé approximants [20,21], symbolic regression [30,31] and although not used directly, it is inspired by the Wright  $\omega$ -function, a cognate of the Lambert W-function [32]. To avoid detailed explanations about the Lambert W-function [33], here it should be noted that in this context it is used to transform the Colebrook equation from the shape implicitly given in respect to the unknown flow friction factor to the explicit form [23,24,27–29,34,35].

### 2.1. Padé Approximants

The ratio of two power series with properly chosen coefficients of the numerator and denominator can approximate very accurately various functions in a narrow zone around the chosen expanding point. For the expressions in the numerator and denominator, Padé approximants [20,21] use rational functions of given order instead of serial expansions. So, in other words, the Padé approximants can estimate functions usually in a narrow zone as the quotient of two polynomials, often has better approximation properties compared with its truncated Taylor series. Being a quotient, the Padé approximants are composed of lower-degree polynomials, where the degree of polynomials can be chosen according to needs. We use Matlab 2019a in order to generate the needed Padé approximants as replacements of the logarithmic function in our rational approximation approach. We do not use expressions with non-integer exponents, because according to Clamond [10], in the software interpretation it is evaluated through one exponential and one logarithmic function (for example  $B^\kappa = e^{\kappa \cdot \ln(B)}$ , where  $\kappa$  is in most cases a non-integer). The computational complexity of an algorithm describes the amount of resources required to run it, for example the execution time. Winning and Coole [13] performed 100 million calculations for each mathematical operation using random inputs, with each repeated five times, and found that the most efficient operation for addition requires 23.4 s.

According to them, relative effort for computation referred to addition-1 as a reference for the following values: logarithm to base, 10–3.37; fractional exponential, 3.32, cubed root, 2.71; natural logarithm, 2.69; cubed, 2.38; square root, 2.29; squared, 2.18; multiplication, 1.55, division, 1.35; subtraction, 1.18.

### 2.2. Symbolic Regression

Symbolic regression is a machine learning approach for finding approximate functions based on evolutionary or genetic algorithms [36]. To avoid imposing prior assumptions on the model, symbolic regression has the ability to search through the space of mathematical expressions to look for an approximate function that best fits a given dataset [31].

We used Eureqa [30], a symbolic regression engine, to obtain our final model. HeuristicLab [37], a software environment for heuristic and evolutionary algorithms, including symbolic regression, can be used instead.

## 3. Routine Based on Polynomial-Form Expressions

### 3.1. Replacement of Logarithmic Function

The Colebrook equation can be accurately approximated using a rational approximation procedure, as shown in Equation (2):

$$\frac{1}{\sqrt{\lambda}} \approx -0.8686 \cdot (\zeta_1 + \zeta_2) \tag{2}$$

where:

$$\zeta_1 = 0.02087 \cdot r - 0.07659 \cdot p(r) - \frac{0.5994}{p(r) + 3.846} - \frac{0.0007232}{r} - 0.00007489 \cdot r^2 + 0.1391$$

$$\zeta_2 = p(r) - 7.93$$

$$p(r) = \frac{r \cdot (r \cdot (11 \cdot r + 27) - 27) - 11}{r \cdot (r \cdot (3 \cdot r + 27) + 27) + 3}$$

$$r = 2777.77 \cdot \left( \frac{2.51 \cdot p_0}{Re} + \frac{\varepsilon}{3.71} \right)$$

$$p_0 = \frac{2600 \cdot Re}{657.7 \cdot Re + 214600 \cdot Re \cdot \varepsilon + 12970000} - 13.58 \cdot \varepsilon + \frac{0.0001165 \cdot Re}{0.00002536 \cdot Re + Re \cdot \varepsilon + 105.5} + 4.227$$

Consequently, Equation (2) contains only rational functions, where:

$\zeta_1 + \zeta_2$  rational approximation of  $\ln\left(\frac{2.51}{Re} \cdot \frac{1}{\sqrt{\lambda}} + \frac{\varepsilon}{3.71}\right)$ ;

$\zeta_1$  a rational function that corrects error caused by Padé approximant  $p(r)$ ;

$\zeta_2$  shifted Padé approximant  $p(r)$ , where the shift  $-7.93 \approx \ln(0.00036)$  where  $0.00036 \approx \frac{1}{2777.77}$ ;

$r$  argument of  $p(r)$ ;

$p(r)$  Padé approximant of  $\ln(r)$  of order /2,3/ at the expansion point  $r = 1$ ;

$p_0$  starting point;

and where:  $\frac{-2}{\ln(10)} \approx \frac{-2}{2.302585093} \approx -0.868$ , as  $\log_{10}(\zeta) = \frac{\ln(\zeta)}{\ln(10)}$ .

Function  $\zeta_2$  approximates the required logarithmic function  $\ln\left(\frac{2.51}{Re} \cdot \frac{1}{\sqrt{\lambda}} + \frac{\varepsilon}{3.71}\right)$ , while  $\zeta_1$  corrects its error, as  $r$  is not always close to the expansion point, because of the large variability of input parameters of the Colebrook equation (Equation (1)). The rational function  $\zeta_1$  and also the starting point  $p_0$  were found by symbolic regression software Eureqa [31], whereas the shift  $-7.93$  in  $\zeta_2$  was found in order to minimize the error of the Padé approximation of  $p(r) \approx \ln(r)$  for the Colebrook equation, as  $\ln(0.00036 \cdot r) \approx \ln(r) - 7.93$ , where  $0.00036 \approx \frac{1}{2777.77}$ . Variable precision arithmetic (VPA) at 4 decimal digit accuracy is assumed for  $\zeta_1$  and for  $p_0$ . The Padé approximant  $p(r)$  is given in

Horner nested polynomial form generated in Matlab 2019a. It is of order /2,3/, which means that the polynomial in numerator contains a monomial of highest degree 2, while the denominator of degree 3. Any other suitable Padé polynomial of any other degree in any other form that can substitute the natural logarithm around the needed expansion point can be used, but in such cases, the rational approximation procedure should be tested again, because these changes can affect the value of the final relative error and its distribution.

### 3.2. Error Analysis

Distribution of the relative error for the proposed rational approximation procedure, Equation (2), is given in Figure 1. The maximal relative error goes up to 0.866% for  $0 < \epsilon < 0.05$  and  $4000 < Re < 10^8$  (as used in engineering practice). The highest error using 2 million input pairs found by the Sobol quasi-Monte Carlo method is for  $Re = 71987$  and  $\epsilon = 3.1711 \cdot 10^{-7}$  [38]. The Colebrook equation is empirical and it follows logarithmic law, so for the procedure with only rational functions, this level of error is acceptable [39]. For example, Pimenta et al. [7] classify the approximation of Sonnad and Goudar [40] with relative error of up to 3.17%. With two logarithms and one non-integer power, this method [40] belongs to the group of approximations with higher performance indexes and precision. Brkić [6] estimates the relative error of Sonnad and Goudar [40] to be up to 0.8%, which is similar error compared with the rational approximation approach presented here; the same methodology as in Brkić [6] is used for Figure 1.

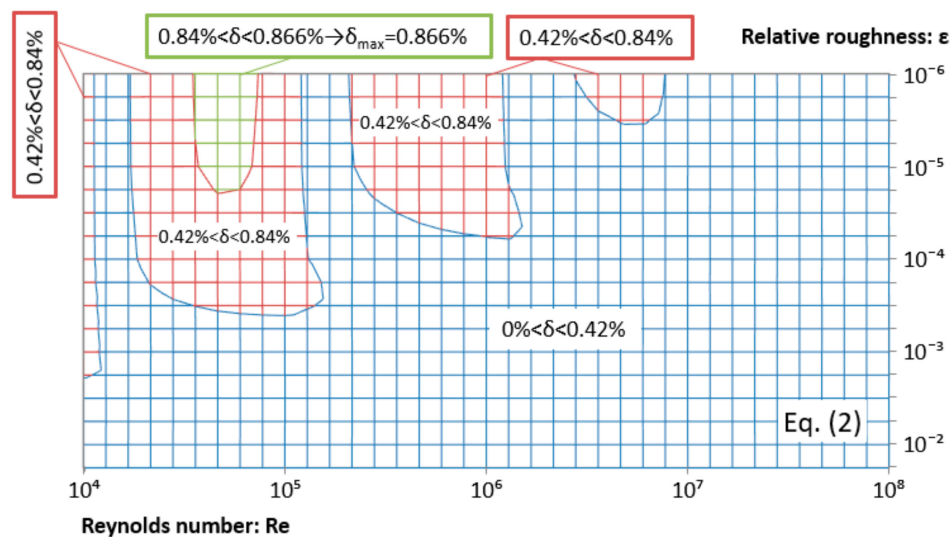


Figure 1. The distribution of the relative error for the proposed rational approximation.

The error in the proposed rational approximation can be possibly reduced by optimizing numerical values of parameters using a genetic algorithms [41] approach with the methodology described by Brkić and Čojbašić [42]. However, the presented rational approximation approach has too many numerical parameters, meaning such an optimization would be very complex. Further simplifications rather should go in the direction of simplification of  $p_0$ ,  $\zeta_1$  and  $\zeta_2$ , but keeping the same or increasing accuracy.

### 3.3. Computational Costs

The efficiency of the proposed procedure is tested using 2 million input pairs found by the Sobol quasi-Monte Carlo method [38]. The tests were performed using Matlab R2019a. The tests revealed that Equation (2) needs 0.56 s to calculate the friction factor  $\lambda$  for 2 million input pairs, or for the Reynolds number  $Re$  and the roughness of the inner pipe surface  $\epsilon$ . On the other hand, the exact solution given by the Wright  $\omega$ -function [23] implemented by the Matlab library “wrightOmegaq” [43] took 1.1 s for the same 2 million the tested pairs.

Our novel rational approximation of the Colebrook equation given with Equation (2) is approximately two times faster than the exact approach using the Wright  $\omega$ -function [23], as the speed ratio is 1.1/0.56–1.96. On the other hand, the approach with the Wright  $\omega$ -function [23] gives the exact solution, which requires two logarithms and one Wright  $\omega$ -function, while this communication presents a rational approximation.

#### 4. Software Description

The presented rational approximation approach for solving Colebrook’s equation for flow friction was thoroughly tested at IT4Innovations, National Supercomputing Center, VŠB-Technical University of Ostrava, Czech Republic.

##### 4.1. Algorithm

The simple algorithm of the rational approximation of the Colebrook equation is presented in Figure 2. The algorithm contains only one branch and is without loops.

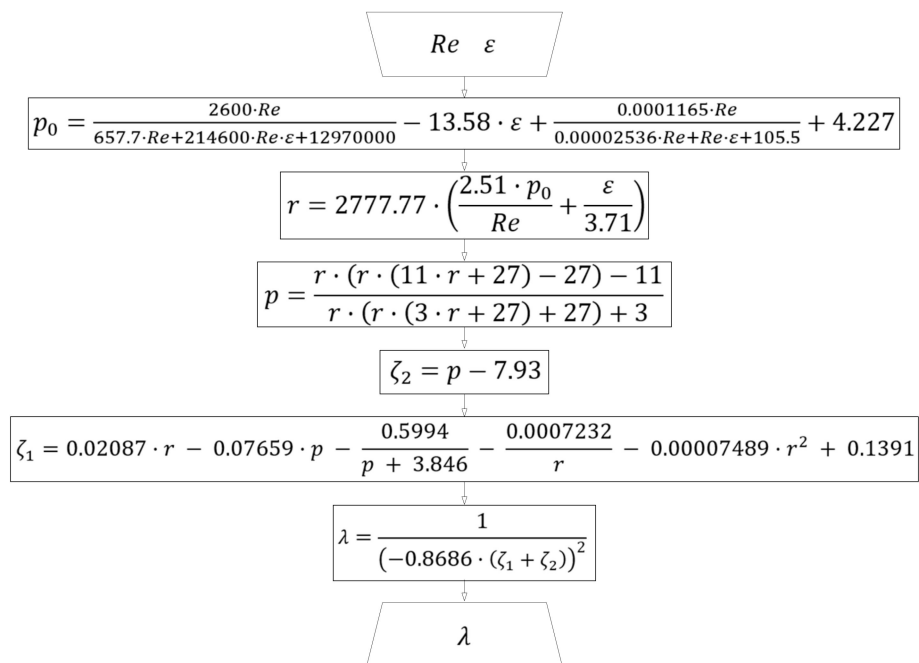


Figure 2. Algorithm of the proposed rational approximation of the Colebrook equation.

##### 4.2. Open-Source Software Code

The code is given in Matlab format, which is compatible with the freeware GNU Octave, but it can be easily transposed in any programming language (input parameters: R is the Reynolds number Re, K is the relative roughness of inner pipe surface  $\epsilon$ ; output parameter; L is the Darcy flow friction factor  $\lambda$ ):

```

x = (2600*R)/(657.7*R + 214600*R.*K + 1.297e+7)-13.58*K
+ (1.165e-4*R)/(2.536e-5*R + R.*K + 105.5) + 4.227
y0 = 2.51*x./R + K./3.71; r = y0*2777.77
pr = (r.*(r.*(11*r + 27) - 27) - 11)/(r.*(r.*(3*r + 27) + 27) + 3)
k1 = @(r) 0.02087*r- 0.07659*pr- 0.5994/(pr + 3.846) - 7.232e-4./r-7.489e-5*r.^2 + 0.1391
x = -0.8686*(k1(r) + pr - 7.93)
L = 1/x.^2
    
```

## 5. Conclusions

We provide a novel rational approximation procedure for solving the logarithmic Colebrook equation for flow friction. Instead of transcendental functions (logarithms, non-integer power) that are used in the classical approach, in this communication, we replace the logarithm with its Padé approximant and a simple rational function, which was found using artificial intelligence (symbolic regression), in order to minimize the error. Although the new rational approximation may seem unintelligible to human eyes, results of 2 million input pairs found by the quasi-Monte Carlo method [38] confirm that the relative error of this new approximation does not exceed 0.866%, which is acceptable for the empirical Colebrook law [44] (trade-off between model complexity and accuracy [45,46]). Consequently, numerical experiments on 2 million of quasi-Monte Carlo pairs indicates that the rational approximation presented here provides for Colebrook's flow friction model a useful combination of Padé approximants and artificial intelligence (symbolic regression).

**Author Contributions:** P.P. got the idea for the presented rational approximation approach and developed its first version. D.B. put the rational approximation approach into a form suitable for everyday engineering use. D.B. made a draft of this short note and prepared figures. P.P. wrote the software code in consultations with D.B. and tested it. All authors have read and agreed to the published version of the manuscript.

**Funding:** The work of P.P. has been partially supported by the Technology Agency of the Czech Republic under the project "National Centre for Energy" TN01000007 and "Energy System for Grids" TK02030039. The work of D.B. has been partially supported by the Ministry of Education, Youth, and Sports of the Czech Republic through the National Programme of Sustainability II (NPS II), IT4Innovations Excellence in Science project LQ1602, and by the Ministry of Education, Science, and Technological Development of the Republic of Serbia under the project "Development of new information and communication technologies, based on advanced mathematical methods, with applications in medicine, telecommunications, power systems, protection of national heritage, and education" iii44006.

**Acknowledgments:** We especially thank Marcelo Masera for his scientific supervision and approval. We acknowledge support from the European Commission, Joint Research Centre (JRC), Directorate C: Energy, Transport, and Climate, Unit C3: Energy Security, Distribution, and Markets.

**Conflicts of Interest:** The authors declare no conflict of interest. Neither the European Commission, the VŠB, Technical University of Ostrava, the research and development Center "Alfatec", nor any person acting on their behalf is responsible for any use of this publication.

## Notations

The following symbols are used in this Communication:

$\lambda$	Darcy, Darcy–Weisbach, Moody, or Colebrook flow friction factor (dimensionless)
$Re$	Reynolds number, $4000 < Re < 10^8$ (dimensionless)
$\varepsilon$	relative roughness of inner pipe surface, $0 < \varepsilon < 0.05$ (dimensionless)
$\zeta_1 + \zeta_2$	rational approximation of $\ln\left(\frac{2.51}{Re} \cdot \frac{1}{\sqrt{\lambda}} + \frac{\varepsilon}{3.71}\right)$
$\zeta_1$	a rational function that corrects error caused by Padé approximant $p(r)$
$\zeta_2$	shifted Padé approximant $p(r)$
$r$	argument of $p(r)$
$p(r)$	Padé approximant of $\ln(r)$ at the expansion point $r$
$p_0$	polynomial starting point
$\log_{10}$	logarithm with base 10
$\ln$	natural logarithm
$e$	exponential function
$\omega$	Wright $\omega$ -function (Wright omega function)

## References

1. Colebrook, C.F. Turbulent flow in pipes with particular reference to the transition region between the smooth and rough pipe laws. *J. Inst. Civ. Eng.* **1939**, *11*, 133–156. [CrossRef]
2. Colebrook, C.; White, C. Experiments with fluid friction in roughened pipes. *Proc. R. Soc. Lond. Ser. A Math. Phys. Sci.* **1937**, *161*, 367–381. [CrossRef]



3. McKeon, B.J.; Zagarola, M.V.; Smits, A.J. A new friction factor relationship for fully developed pipe flow. *J. Fluid Mech.* **2005**, *538*, 429–443. [[CrossRef](#)]
4. Brkić, D.; Praks, P. Unified friction formulation from laminar to fully rough turbulent flow. *Appl. Sci.* **2018**, *8*, 2036. [[CrossRef](#)]
5. Moody, L.F. Friction factors for pipe flow. *Trans. ASME* **1944**, *66*, 671–684.
6. Brkić, D. Review of explicit approximations to the Colebrook relation for flow friction. *J. Pet. Sci. Eng.* **2011**, *77*, 34–48. [[CrossRef](#)]
7. Pimenta, B.D.; Robaina, A.D.; Peiter, M.X.; Mezzomo, W.; Kirchner, J.H.; Ben, L.H.B. Performance of explicit approximations of the coefficient of head loss for pressurized conduits. *Rev. Bras. Eng. Agríc. Ambient.* **2018**, *22*, 301–307. [[CrossRef](#)]
8. Praks, P.; Brkić, D. Advanced iterative procedures for solving the implicit Colebrook equation for fluid flow friction. *Adv. Civ. Eng.* **2018**, *2018*, 5451034. [[CrossRef](#)]
9. Praks, P.; Brkić, D. Choosing the optimal multi-point iterative method for the Colebrook flow friction equation. *Processes* **2018**, *6*, 130. [[CrossRef](#)]
10. Alfaro-Guerra, M.; Guerra-Rojas, R.; Olivares-Gallardo, A. Evaluación experimental de la solución analítica exacta de la ecuación de Colebrook-White (Experimental evaluation of exact analytical solution of the Colebrook-White Equation). *Ing. Investig. Tecnol.* **2019**, *20*, 1–11. (in Spanish). [[CrossRef](#)]
11. Clamond, D. Efficient resolution of the Colebrook equation. *Ind. Eng. Chem. Res.* **2009**, *48*, 3665–3671. [[CrossRef](#)]
12. Winning, H.K.; Coole, T. Explicit friction factor accuracy and computational efficiency for turbulent flow in pipes. *Flow Turbul. Combust.* **2013**, *90*, 1–27. [[CrossRef](#)]
13. Winning, H.K.; Coole, T. Improved method of determining friction factor in pipes. *Int. J. Numer. Methods Heat Fluid Flow* **2015**, *25*, 941–949. [[CrossRef](#)]
14. Biberg, D. Fast and accurate approximations for the Colebrook equation. *J. Fluids Eng.* **2017**, *139*, 031401. [[CrossRef](#)]
15. Vatankhah, A.R. Approximate analytical solutions for the Colebrook equation. *J. Hydraulic Eng.* **2018**, *144*, 06018007. [[CrossRef](#)]
16. Brkić, D.; Praks, P. Discussion of “Approximate analytical solutions for the Colebrook equation”. *J. Hydraulic Eng.* **2020**, *146*, 07019011. [[CrossRef](#)]
17. Lamri, A.A. Discussion of “Approximate analytical solutions for the Colebrook equation”. *J. Hydraulic Eng.* **2020**, *146*, 07019011. [[CrossRef](#)]
18. Vatankhah, A.R. Closure to “Approximate analytical solutions for the Colebrook equation”. *J. Hydraulic Eng.* **2020**, *146*, 07019013. [[CrossRef](#)]
19. Praks, P.; Brkić, D. One-log call iterative solution of the Colebrook equation for flow friction based on Padé polynomials. *Energies* **2018**, *11*, 1825. [[CrossRef](#)]
20. Baker, G.A.; Graves-Morris, P. *Padé Approximants*, 2nd ed.; Cambridge University Press: Cambridge, UK, 1996.
21. Copley, L. *Padé Approximants, Mathematics for the Physical Sciences*. *Walter Gruyter GmbH Co KG* **2014**, *7*, 163–179. [[CrossRef](#)]
22. Brkić, D.; Praks, P. Colebrook’s flow friction explicit approximations based on fixed-point iterative cycles and symbolic regression. *Computation* **2019**, *7*, 48. [[CrossRef](#)]
23. Brkić, D.; Praks, P. Accurate and efficient explicit approximations of the Colebrook flow friction equation based on the Wright  $\omega$ -function. *Mathematics* **2019**, *7*, 34. [[CrossRef](#)]
24. Brkić, D.; Praks, P. Accurate and efficient explicit approximations of the Colebrook flow friction equation based on the Wright  $\omega$ -function: Reply to Discussion. *Mathematics* **2019**, *7*, 410. [[CrossRef](#)]
25. Muzzo, L.E.; Pinho, D.; Lima, L.E.; Ribeiro, L.F. Accuracy/Speed analysis of pipe friction factor correlations. In *International Congress on Engineering and Sustainability in the XXI Century—INCREaSE 2019, Section: Water for Ecosystem and Society, Faro, Portugal, 9–11 October 2019*; Monteiro, J., Silva, A.J., Mortal, A., Aníbal, J., da Silva, M.M., Oliveira, M., Eds.; Springer Nature Switzerland AG 2020: Cham, Switzerland, 2019; pp. 664–679. [[CrossRef](#)]
26. Praks, P.; Brkić, D. Symbolic regression-based genetic approximations of the Colebrook equation for flow friction. *Water* **2018**, *10*, 1175. [[CrossRef](#)]
27. Brkić, D. W solutions of the CW equation for flow friction. *Appl. Math. Lett.* **2011**, *24*, 1379–1383. [[CrossRef](#)]



28. Brkić, D. Lambert W function in hydraulic problems. *Math. Balk.* **2012**, *26*, 285–292. Available online: <http://www.math.bas.bg/infres/MathBalk/MB-26/MB-26-285-292.pdf> (accessed on 13 November 2019).
29. Brkić, D. Discussion of “Exact analytical solutions of the Colebrook-White equation” by Yozo Mikata and Walter, S. Walczak. *J. Hydraul. Eng.* **2017**, *143*, 07017007. [[CrossRef](#)]
30. Dubčáková, R. Eureka: Software review. *Genet. Program. Evolvable Mach.* **2011**, *12*, 173–178. [[CrossRef](#)]
31. Schmidt, M.; Lipson, H. Distilling free-form natural laws from experimental data. *Science* **2009**, *324*, 81–85. [[CrossRef](#)]
32. Hayes, B. Why W? *Am. Sci.* **2005**, *93*, 104–108. [[CrossRef](#)]
33. Corless, R.M.; Gonnet, G.H.; Hare, D.E.; Jeffrey, D.J.; Knuth, D.E. On the Lambert W function. *Adv. Comput. Math.* **1996**, *5*, 329–359. [[CrossRef](#)]
34. Brkić, D. An explicit approximation of Colebrook’s equation for fluid flow friction factor. *Pet. Sci. Technol.* **2011**, *29*, 1596–1602. [[CrossRef](#)]
35. Brkić, D. Comparison of the Lambert W-function based solutions to the Colebrook equation. *Eng. Comput.* **2012**, *29*, 617–630. [[CrossRef](#)]
36. Giustolisi, O.; Savić, D.A. A symbolic data-driven technique based on evolutionary polynomial regression. *J. Hydroinformatics* **2006**, *8*, 207–222. [[CrossRef](#)]
37. Wagner, S.; Kronberger, G.; Beham, A.; Kommenda, M.; Scheibenpflug, A.; Pitzer, E.; Vonolfen, S.; Kofler, M.; Winkler, S.; Dorfer, V.; et al. Architecture and design of the HeuristicLab optimization environment. *Top. Intell. Eng. Inform.* **2014**, *6*, 197–261. [[CrossRef](#)]
38. Sobol, I.M.; Turchaninov, V.I.; Levitan, Y.L.; Shukhman, B.V. *Quasi-Random Sequence Generators*; Distributed by OECD/NEA Data Bank; Keldysh Institute of Applied Mathematics; Russian Academy of Sciences: Moscow, Russia, 1992; Available online: <https://ec.europa.eu/jrc/sites/jrcsh/files/LPTAU51.rar> (accessed on 19 November 2019).
39. Eck, B.J.; Mevissen, M. Quadratic approximations for pipe friction. *J. Hydroinformatics* **2015**, *17*, 462–472. [[CrossRef](#)]
40. Sonnad, J.R.; Goudar, C.T. Turbulent flow friction factor calculation using a mathematically exact alternative to the Colebrook–White equation. *J. Hydraul. Eng.* **2006**, *132*, 863–867. [[CrossRef](#)]
41. Mitrev, R.; Tudjarov, B.; Todorov, T. Cloud-based expert system for synthesis and evolutionary optimization of planar linkages. *Facta Univ. Ser. Mech. Eng.* **2018**, *16*, 139–155. [[CrossRef](#)]
42. Brkić, D.; Čojbašić, Ž. Evolutionary optimization of Colebrook’s turbulent flow friction approximations. *Fluids* **2017**, *2*, 15. [[CrossRef](#)]
43. Horchler, A.D. Complex Double-Precision Evaluation of the Wright Omega Function. 2017. Available online: <https://github.com/horchler/wrightOmegaq> (accessed on 10 December 2019).
44. Niazkar, M. Revisiting the estimation of Colebrook friction factor: A comparison between artificial intelligence models and CW based explicit equations. *KSCE J. Civ. Eng.* **2019**, *23*, 4311–4326. [[CrossRef](#)]
45. Davidson, J.W.; Savić, D.; Walters, G.A. Method for the identification of explicit polynomial formulae for the friction in turbulent pipe flow. *J. Hydroinformatics* **1999**, *1*, 115–126. [[CrossRef](#)]
46. Giustolisi, O.; Berardi, L.; Walski, T.M. Some explicit formulations of Colebrook–White friction factor considering accuracy vs. computational speed. *J. Hydroinformatics* **2010**, *13*, 401–418. [[CrossRef](#)]

