data

MDPI

Data Descriptor

Sampling the Darcy Friction Factor Using Halton, Hammersley, Sobol, and Korobov Sequences: Data Points from the Colebrook Relation

Dejan Brkić [1,2,3,*] and Marko Milošević [1]

[1] Faculty of Electronic Engineering, University of Niš, 18000 Niš, Serbia; marko.milosevic@elfak.ni.ac.rs
[2] University of Belgrade, 11000 Beograd, Serbia
[3] IT4Innovations, VSB-Technical University of Ostrava, 708 00 Ostrava, Czech Republic
[*] Correspondence: dejan.brkic@elfak.ni.ac.rs or dejanrgf@tesla.rcub.bg.ac.rs or dejan.brkic@vsb.cz

Abstract

When the Colebrook equation is used in its original implicit form, the unknown pipe flow friction factor can only be obtained through time-consuming and computationally demanding iterative calculations. The empirical Colebrook equation relates the unknown Darcy friction factor to a known Reynolds number and a known relative roughness of a pipe's inner surface. It is widely used in engineering. To simplify computations, a variety of explicit approximations have been developed, the accuracy of which must be carefully evaluated. For this purpose, this Data Descriptor gives a sufficient number of pipe flow friction factor values that are computed using a highly accurate iterative algorithm to solve the implicit Colebrook equation. These values serve as reference data, spanning the range relevant to engineering applications, and provide benchmarks for evaluating the accuracy of the approximations. The sampling points within the datasets are distributed in a way that minimizes gaps in the data. In this study, a Python Version v1 script was used to generate quasi-random samples, including Halton, Hammersley, Sobol, and deterministic lattice-based Korobov samples, which produce smaller gaps than purely random samples generated for comparison purposes. Using these sequences, a total of 2^{20} = 1,048,576 data points were generated, and the corresponding datasets are provided in in the zenodo repositoryWhen a smaller subset of points is needed, the required number of initial points from these sequences can be used directly.

Dataset: https://doi.org/10.5281/zenodo.17280142

Dataset License: CC-BY 4.0

Keywords: Colebrook formula; explicit approximations; Darcy friction factor; pipe liquid flow; Halton distribution; Hammersley distribution; Sobol distribution; Korobov distribution; Colebrook–White experiment; Moody chart

check for updates

Academic Editor: André Bauer

Received: 6 October 2025
Revised: 3 November 2025
Accepted: 12 November 2025
Published: 20 November 2025

Citation: Brkić, D.; Milošević, M. Sampling the Darcy Friction Factor Using Halton, Hammersley, Sobol, and Korobov Sequences: Data Points from the Colebrook Relation. *Data* **2025**, *10*, 193. https://doi.org/10.3390/data10110193

1. Summary

The empirical Colebrook equation [1] is an implicit empirical relation for turbulent flow that relates the unknown Darcy friction factor λ to a known Reynolds number Re and a known relative roughness of a pipe's inner surface ε/D. It is widely used in engineering. Its primary limitation is its implicit nature, which requires iterative numerical methods for solution rather than a direct analytical expression. To provide reliable reference values for engineering applications, comprehensive datasets based on the Colebrook equation were generated and presented in this Data Descriptor. They are intended to serve as benchmarks for assessing the accuracy of numerous explicit approximations of the Colebrook equation and can also serve as a computational substitute for the Moody chart. These datasets cover the practically relevant parameter space encountered in engineering, spanning Reynolds numbers (Re) from 4000 to 10^8 and relative pipe roughness values (ε/D) from 0 to 0.05.

This Data Descriptor answers the following questions:

1. How were the data points generated? To obtain accurate values, a highly precise iterative solver of the Colebrook equation was applied. In this way, the datasets were generated by computing the unknown Darcy friction factor λ from a known Reynolds number Re and a known roughness of a pipe's inner surface ε/D—(Re, ε/D)→λ.
2. How are the data points distributed? To minimize gaps in coverage, the data points were distributed using sampling methods. Specifically, Halton, Hammersley, Sobol, and Korobov sequences were employed. The Halton, Hammersley, and Sobol methods are quasi-random (low-discrepancy) techniques, while the Korobov method is a deterministic lattice-based approach. For each of these four sequences, 2^{20} samples were generated, yielding 1,048,576 friction factor data points for each method. When a smaller subset of points is needed, the required number of initial points from these large sequences can be used directly.

Appendix A provides a Python script that can be used to generate Halton, Hammersley, Sobol, and Korobov samples normalized for the Colebrook equation—$\{S_j \rightarrow \mathrm{Re}, S_k \rightarrow \varepsilon/D\} \rightarrow \lambda$.

2. Data Description

2.1. Generation of the Data Points

The empirical Colebrook relation, given in Equation (1), dating from the 1930s [1], is widely accepted in engineering as an informal standard for calculating the flow friction factor in pipes:

$$\frac{1}{\sqrt{\lambda}} = -0.8686 \cdot \ln\left(\frac{2.51}{\mathrm{Re}} \cdot \frac{1}{\sqrt{\lambda}} + \frac{1}{3.71} \cdot \frac{\varepsilon}{\mathrm{D}}\right), \tag{1}$$

where

λ is the Darcy friction factor (dimensionless);
Re is the Reynolds number (dimensionless);
D is the inner diameter of a pipe (meters);
ε/D is the relative roughness of a pipe's inner surface (dimensionless);
ln stands for natural logarithm.

The Colebrook equation is based on an experiment performed by Colebrook and White [2] regarding the flow of a fluid through sets of pipes with different levels of roughness (starting from smooth).

Unfortunately, the Colebrook equation is an implicit equation for the unknown friction factor λ, with λ appearing on both sides of the equals sign in a logarithmic form from which it cannot be extracted or transformed in explicit form without approximations (an exception is the Lambert W-function [3–8]—further evaluation of the Lambert W-function is also approximate but highly accurate when using various mathematical approaches to evaluate this function). Many such approximations have been developed, and they can be compared with respect to their accuracy and complexity [9–24].

Therefore, an iterative procedure, which can achieve practically any desired accuracy, is used in this Data Descriptor to compute the highly accurate values of the points employed in constructing the presented datasets available at https://doi.org/10.5281/zenodo.17280142. The iterative solution, using recommendations from this Data Descriptor, is treated as accurate, and therefore the Colebrook equation uses the sign '=', while its approximations use '≈'.

To start iterations, the initial step is estimated by omitting the $\frac{2.51}{\mathrm{Re}}\cdot\frac{1}{\sqrt{\lambda}}$ term from Equation (1), yielding a reasonable initial guess, λ_0, which is based only on the roughness of the inner surface of a pipe, ε, which is valid for a fully developed rough turbulent regime λ_∞, as given in Equation (2):

$$\frac{1}{\sqrt{\lambda_0}} = \frac{1}{\sqrt{\lambda_\infty}} \approx -0.8686\cdot\ln\left(\frac{1}{3.71}\cdot\frac{\varepsilon}{\mathrm{D}}\right) \tag{2}$$

The fixed-point iterative process [25] continues with the calculation of the values λ_1 (a function of λ_0), λ_2 (a function of λ_1), λ_i (a function of λ_{i-1}),$\cdots$, λ_n (a function of λ_{n-1}) using Equation (1), where n typically ranges from a few iterations up to 100 iterations to reach an accuracy of 10^{-9} (for highly accurate engineering applications, a precision of 10^{-6} is sufficient). However, given the empirical nature of the Colebrook equation, an accuracy of about two significant digits is adequate for most calculations, while additional digits beyond that are essentially numerical noise.

A limitation in accuracy can be noticed in cases where the roughness ε is very small (approaching zero) when the logarithm becomes problematic, leading to precision issues. Measurement and evaluation of pipe roughness also introduce a certain level of uncertainty [26,27].

The widely recognized Moody diagram [28–32] offers a graphical representation of the Colebrook relation. Instead of reading from the Moody diagram [33,34], datasets from https://doi.org/10.5281/zenodo.17280142 should be used. These data can also be used to redraw the Moody chart.

Datasets similar to those given here, but with a uniform distribution (1000-by-1000 mesh), are available [35].

2.2. Distributions of the Data Points

The Colebrook equation involves two input parameters: the Reynolds number (Re) and relative roughness of a pipe's inner surface (ε/D). Therefore, a two-dimensional sequence of pairs (S_j, S_k) is required as an input. Using Equation (3) [36], pairs (S_j, S_k), each between 0 and 1, are transformed into the Reynolds numbers Re ($4000 < \mathrm{Re} < 10^8$) and relative roughness of a pipe's inner surface, corresponding to $0 < \varepsilon/\mathrm{D} < 0.05$ ($S_j \rightarrow \mathrm{Re}$ and $S_k \rightarrow \varepsilon/\mathrm{D}$).

$$\left.\begin{aligned} Re_j &= 10^{S_j\cdot(log_{10}(10^8)-log_{10}(4000))+log_{10}(4000)} \\ \varepsilon/D_k &= 10^{-(S_k\cdot(6.5-log_{10}(1/0.05))+log_{10}(1/0.05))} \end{aligned}\right\}, \tag{3}$$

where

$\log_{10}$ denotes the base-10 (Briggs) logarithm.

Figures 1–5 show 2^8 = 256 data points, while Tables A1–A4 of Appendix B show the first 2^2 = 4 data points. The full datasets with 2^{20} = 1,048,576 samples are available at https://doi.org/10.5281/zenodo.17280142. If a smaller dataset is required, one can be obtained by randomly selecting the desired number of data points by uniformly omitting points (e.g., every nth point), or, in some cases, by selecting the first portion of the data points when appropriate—always ensuring that no gaps in data coverage occur. In general, the first subset of points (e.g., 2^{10} = 1024 from 2^{20} = 1,048,576) generated by Halton, Hammersley, Sobol, and Korobov sequences can be used directly, as they already provide good spatial coverage; among them, the Sobol sequence typically yields the most uniform distribution, while the others exhibit only slightly lower uniformity.

Halton, Hammersley, Sobol, and Korobov sequences were specifically chosen to generate the datasets provided at https://doi.org/10.5281/zenodo.17280142. These sequences were employed to generate the points for the Colebrook dataset, ensuring thorough and reproducible coverage of the Reynolds number Re and relative roughness ε/D parameter space. Each method for sampling offers a different balance between uniformity, computational simplicity, and scalability. All are designed to generate well-distributed points in multidimensional space—being applied specifically to two-dimensional datasets for modeling purposes in this study. Unlike random distributions, quasi-random and deterministic lattice-based sequences provide coverage with minimized gaps and produce identical point arrangements across realizations, facilitating proper repetitions and reevaluations of computational experiments (in this Data Descriptor, the focus is on two-dimensional space—$S_i \rightarrow Re$ and $S_k \rightarrow \varepsilon/D$; the distribution of dots in random distribution changes as in Section 2.2.1, while that in Section 2.2.2 remains identical each time in each realization). Their practical value lies in generating well-distributed sampling points in multidimensional spaces; theoretical details on their construction are beyond the scope of this Data Descriptor.

2.2.1. Random Distributions

The random distributions are inherently non-deterministic, resulting in different arrangements of data points across the spaces S_1 and S_2 in each realization (the distribution of dots from Figure 1 will be different in each realization). This can cause certain problems regarding repetitions and reevaluations of the results of computational experiments.

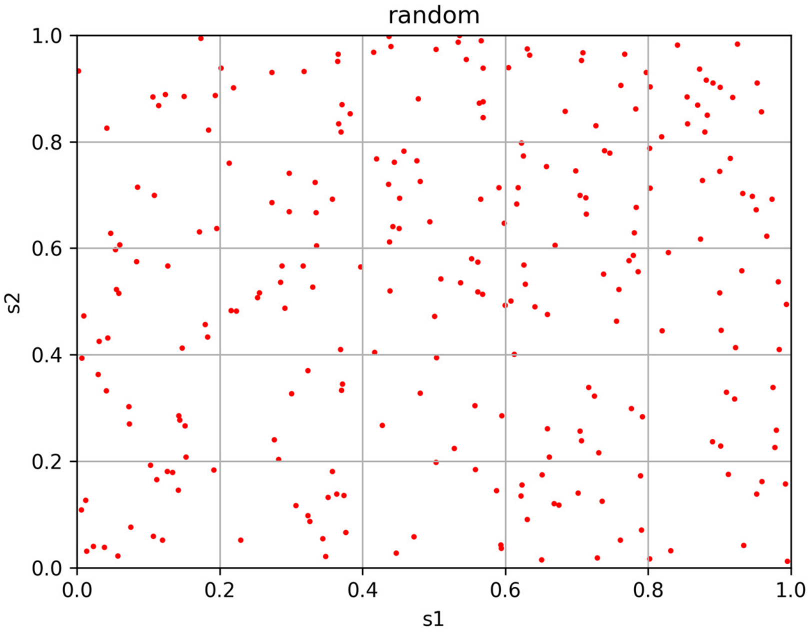

Figure 1. Random distribution.

2.2.2. Quasi-Random Distributions

The Halton [37–40], Hammersley [40,41], and Sobol [42–47] methods are quasi-random (low-discrepancy) techniques designed to generate well-distributed points in multidimensional space—being applied specifically to two-dimensional datasets for modeling purposes in this case.

- Halton Quasi-Random Distribution (Figure 2)

The Halton sample is simple to implement; it provides good, uniform coverage in low-dimensional spaces such as those discussed here, where two dimensions are required (performance deteriorates beyond ~10 dimensions).

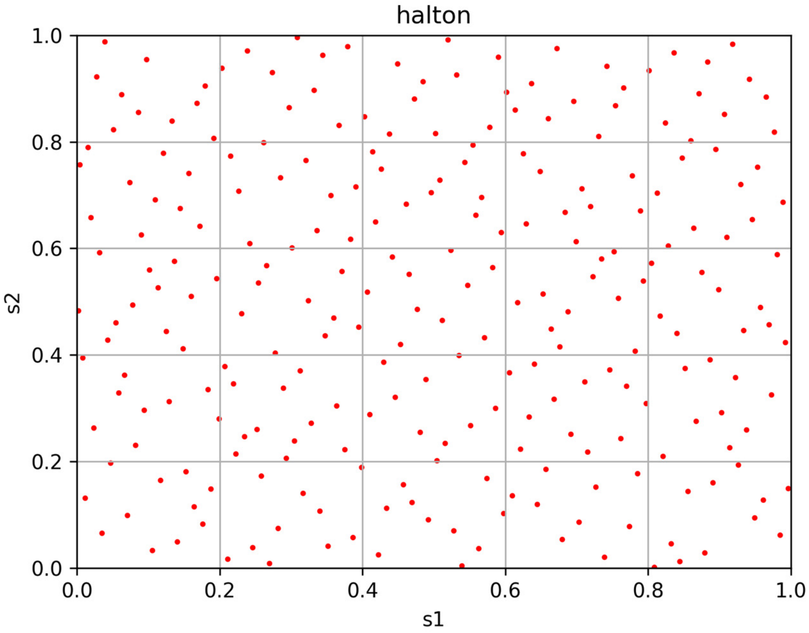

Figure 2. Halton quasi-random distributions.

- Hammersley Quasi-Random Distribution (Figure 3)

The Hammersley sample shows behavior similar to that of the Halton distribution, especially when a finite number of samples is predefined.

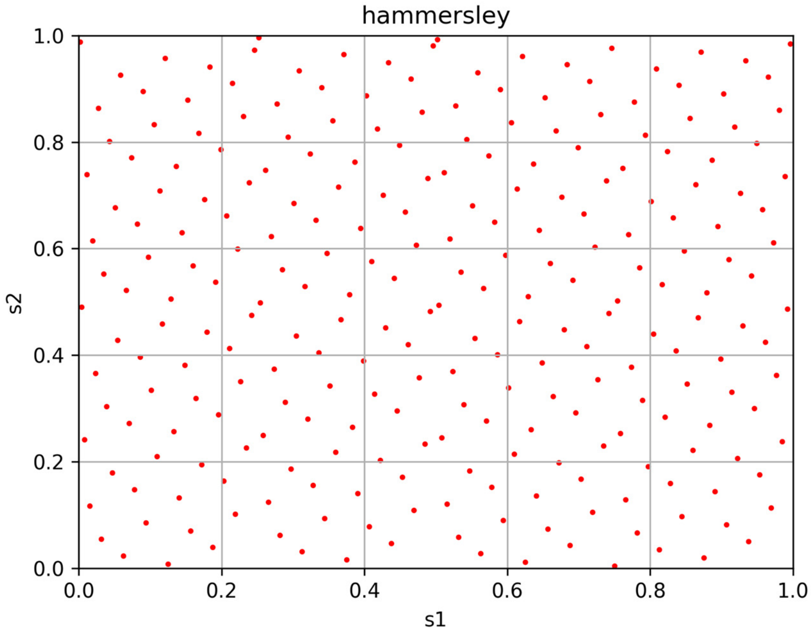

Figure 3. Hammersley quasi-random distributions.

- Sobol Quasi-Random Distribution (Figure 4)

Sobol sequences can be extended dynamically, point by point, without losing uniformity. This characteristic makes them particularly well-suited for selecting a smaller subset of points from a larger dataset simply by truncating the sequence.

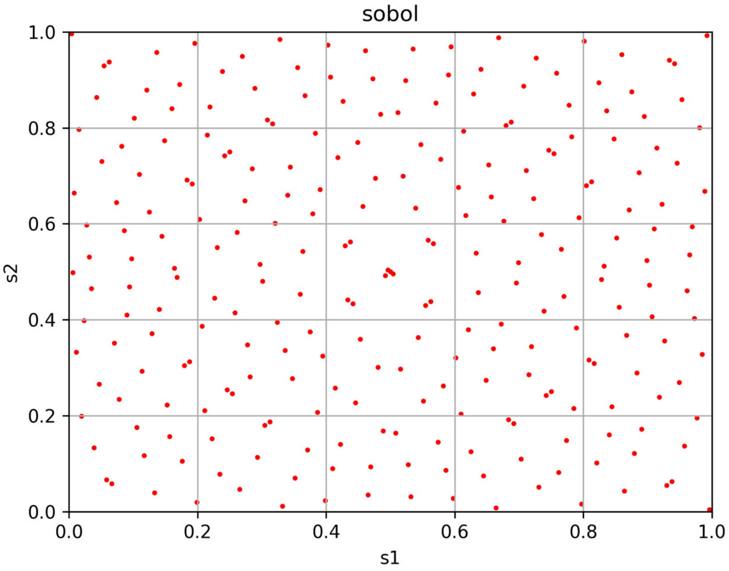

Figure 4. Sobol quasi-random distributions.

2.2.3. Deterministic Lattice-Based Korobov Distribution

The Korobov sequence belongs to the broader family of lattice rules, which are based on modular arithmetic rather than randomization or discrepancy minimization in the quasi-Monte Carlo sense. The Korobov distribution generates structured, grid-like data distributions (which are less effective for non-periodic or irregular domains). While Korobov sequences can also produce evenly distributed points and are used for modeling similarly to quasi-random sequences, they are more accurately described as deterministic lattice-based or number-theoretic methods rather than quasi-random ones. They provide a compromise between uniformity and quasi-random sampling.

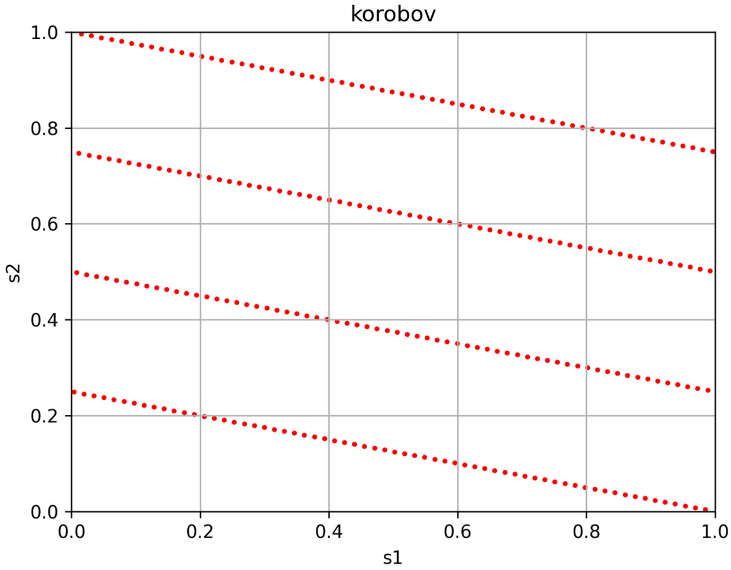

Figure 5. Korobov deterministic lattice-based distribution.

3. Methods: Evaluation of Accuracy Using the Datasets

The relative error $\delta_{\%,jk}$ of the evaluated explicit approximation of the Colebrook equation can be calculated using Equation (4):

$$\delta_{\%,jk} = \frac{\left|\lambda_{apr,jk} - \lambda_{jk}\right|}{\lambda_{jk}} \times 100\%, \tag{4}$$

where

$\delta_{\%,jk}$ is the relative error (dimensionless);

λ_{apr} is the Darcy friction factor (dimensionless) obtained from the observed approximation;

λ is the accurate Darcy friction factor (dimensionless) derived from the datasets (available at https://doi.org/10.5281/zenodo.17280142);

jk is the position of the accurate Darcy friction factor (dimensionless) with respect to the datasets from https://doi.org/10.5281/zenodo.17280142;

$| \; |$ denotes the absolute value.

Maximal relative error across 2^{20} = 1,048,576 quasi-random samples from the datasets from https://doi.org/10.5281/zenodo.17280142 should be identified accordingly.

Practical examples are given in Appendix C.

Author Contributions: Conceptualization, D.B.; methodology, D.B.; software, M.M.; validation, D.B.; formal analysis, D.B.; investigation, D.B.; resources, D.B.; data curation, D.B.; writing—original draft preparation, D.B.; writing—review and editing, D.B.; visualization, D.B. and M.M.; supervision, D.B.; project administration, D.B. and M.M.; funding acquisition, D.B. and M.M. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Ministry of Science, Technological Development, and Innovations of the Republic of Serbia under grant number 451-03-136/2025-03/200102 and by the European Union under the project “Increasing the resilience of power grids in the context of decarbonisation, decentralisation and sustainable socioeconomic development”, CZ.02.01.01/00/23_021/0008759 through the Operational Programme Johannes Amos Comenius.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data are contained within the article, and they are also available at https://doi.org/10.5281/zenodo.17280142.

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

Appendix A

```
import chaospy
import math
import csv
import matplotlib.pyplot as plt
import os

def initial_guess_drop_term(eps):
    arg = eps/3.71
    if arg <= 0:
        raise ValueError(“eps must be > 0”)
    inv_sqrt_f0 = -0.8686 * math.log(arg)
    if inv_sqrt_f0 <= 0:
        raise ValueError(“Computed 1/sqrt(f0) is non-positive (check eps).”)
    f0 = 1.0/(inv_sqrt_f0 * inv_sqrt_f0)
    return f0

def colebrook_refine(Re, eps, tol = 1e-9, max_iter = 100):
    if Re <= 0:
        raise ValueError(“Re must be > 0”)
    f = initial_guess_drop_term(eps)
    for _ in range(max_iter):
        inv_sqrt_f = 1.0/math.sqrt(f)
        arg = (2.51/Re) * inv_sqrt_f + eps/3.71
        if arg <= 0:
            raise ValueError(f“Non-positive argument to log: {arg}”)
        rhs = -0.8686 * math.log(arg)
        f_new = 1.0/(rhs * rhs)

        if abs(f_new - f) < tol:
            return f_new
        f = f_new

    raise RuntimeError(“Colebrook iteration did not converge”)
def calc_Re(s1):
    return 10 ** (s1 * (math.log10(10 ** 8) - math.log10(4000)) + math.log10(4000))

def calc_eps(s2):
    return 10 ** (-(s2 * (6.5 - math.log10(1/0.05)) + math.log10(1/0.05)))

def process_samples(samples, method_name):

    os.makedirs(method_name, exist_ok = True)

    csv_path = os.path.join(method_name, f“{method_name}_results.csv”)
    with open(csv_path, “w”, newline = “”) as csvfile:
        writer = csv.writer(csvfile)
        writer.writerow([“x”, “y”, “Re”, “epsilon”, “f0”])

        for i in range(samples.shape[1]):
            s1, s2 = samples[:, i]
            Re = calc_Re(s1)
            eps = calc_eps(s2)
            f0 = colebrook_refine(Re, eps)
            writer.writerow([s1, s2, Re, eps, f0])

    plt.scatter(*samples, s = 3, c = “red”)
    plt.title(method_name)
    plt.xlabel(“s1”)
    plt.ylabel(“s2”)
    plt.xlim(0, 1)
    plt.ylim(0, 1)
    plt.grid(True)

    plot_path = os.path.join(method_name, f“{method_name}_plot.png”)
    plt.savefig(plot_path, dpi = 300)
    plt.close()

def main():
    uniform_cube = chaospy.J(chaospy.Uniform(0, 1), chaospy.Uniform(0, 1))
    count = 2 ** 20

    samples_dict = {
        “halton”: uniform_cube.sample(count, rule = “halton”),
        “hammersley”: uniform_cube.sample(count, rule = “hammersley”),
        “korobov”: uniform_cube.sample(count, rule = “korobov”),
        “sobol”: uniform_cube.sample(count, rule = “sobol”),
    }

    for method, samples in samples_dict.items():
        process_samples(samples, method)

if __name__ == “__main__”:
    main()
```

Appendix B

The first $2^2 = 4$ data points of the Halton, Hammersley, and Sobol quasi-random sequences and the Korobov deterministic lattice-based sequence for the Colebrook equation are given in Tables A1–A4.

Table A1. First $2^2 = 4$ data points of the Halton quasi-random sequence for the Colebrook equation.

S_1	S_2	**Re**	**ε/D**	**λ**
0.125	0.444444444	14,184.12312	0.000244521	0.028646643
0.625	0.777777778	2,242,706.784	4.52×10^{-6}	0.010356771
0.375	0.222222222	178,355.9058	0.003496579	0.027989349
0.875	0.555555556	28,200,544.83	6.47×10^{-5}	0.011135324

Table A2. First $2^2 = 4$ data points of the Hammersley quasi-random sequence for the Colebrook equation.

S_1	S_2	**Re**	**ε/D**	**λ**
0.75	0.003891051	7,952,707.288	0.047724408	0.069941163
0.125	0.007782101	14,184.12312	0.045552382	0.07014454
0.625	0.011673152	2,242,706.784	0.043479209	0.067049409
0.375	0.015564202	178,355.9058	0.041500391	0.065798735

Table A3. First $2^2 = 4$ data points of the Sobol quasi-random sequence for the Colebrook equation.

S_1	S_2	**Re**	**ε/D**	**λ**
0.5	0.5	632,455.532	0.000125743	0.014350582
0.75	0.25	7,952,707.288	0.002507422	0.024895569
0.25	0.75	50,297.33719	6.31×10^{-6}	0.020886224
0.375	0.375	178,355.9058	0.000561508	0.019315882

Table A4. First $2^2 = 4$ data points of the Korobov deterministic lattice-based sequence for the Colebrook equation.

S_1	S_2	**Re**	**ε/D**	**λ**
0.003891051	0.249027237	4160.759355	0.002536792	0.041979229
0.007782101	0.498054475	4327.979604	0.000128706	0.039126016
0.011673152	0.747081712	4501.920406	6.53×10^{-6}	0.038552156
0.015564202	0.996108949	4682.851862	3.31×10^{-7}	0.038106614

Appendix C

The accuracy and computational complexity of the explicit approximations of the Colebrook formula for flow friction can be practically evaluated using the datasets presented in this Data Descriptor. Recently proposed approximations by Ferreri [12,13] can be used as an illustrative example for this purpose. Ferreri's approximations are further compared with existing formulations from the literature that are highly accurate and computationally efficient.

Appendix C.1. New Ferreri's Approximations

Ferreri [12,13] proposed two explicit approximations of the Colebrook white relation, a simpler Equation (A1) and a more complex Equation (A2):

$$\frac{1}{\sqrt{\lambda}} \approx \frac{1}{\sqrt{\lambda_\infty}} - 1.9 \cdot \log_{10}\left(\frac{3}{\mathrm{Re} \cdot \frac{\varepsilon}{\mathrm{D}} \cdot \sqrt{\frac{\lambda_\infty}{8}}} + 1\right), \tag{A1}$$

$$\left.\begin{array}{r} \frac{1}{\sqrt{\lambda}} \approx -2 \cdot log_{10}\left(\frac{2.51}{Re} \cdot \frac{1}{\sqrt{A \cdot \lambda_\infty}} + \frac{1}{3.71} \cdot \frac{\varepsilon}{D}\right) \\ A \approx 1 + 0.0066 \cdot \left(\frac{\varepsilon}{D}\right)^{-0.203} \cdot \left[log_{10}\left(\frac{0.86}{\left(\frac{\varepsilon}{D}\right)^{0.0985}}\right)\right]^{log_{10}\left(\frac{Re}{70} \cdot \frac{\varepsilon}{D} \cdot \sqrt{\frac{\lambda_\infty}{8}}\right)} \end{array}\right\}, \tag{A2}$$

The first 2^{10} = 1024 Sobol points from this Data Descriptor are used for the evaluation of accuracy in these two cases. The maximal relative error of Ferreri's approximations is computed using Equation (4), and it is 3.25% for the simpler expression, which is given in Equation (A1), and 2.45% for the more complex version, which is given in Equation (A2). The distribution of the relative error δ% for the simpler version of Ferreri's explicit approximation is given in Figure A1, while for the more complex version, the distribution is in Figure A2. The methodology used to plot Figures A1 and A2 is the same as that used in [9].

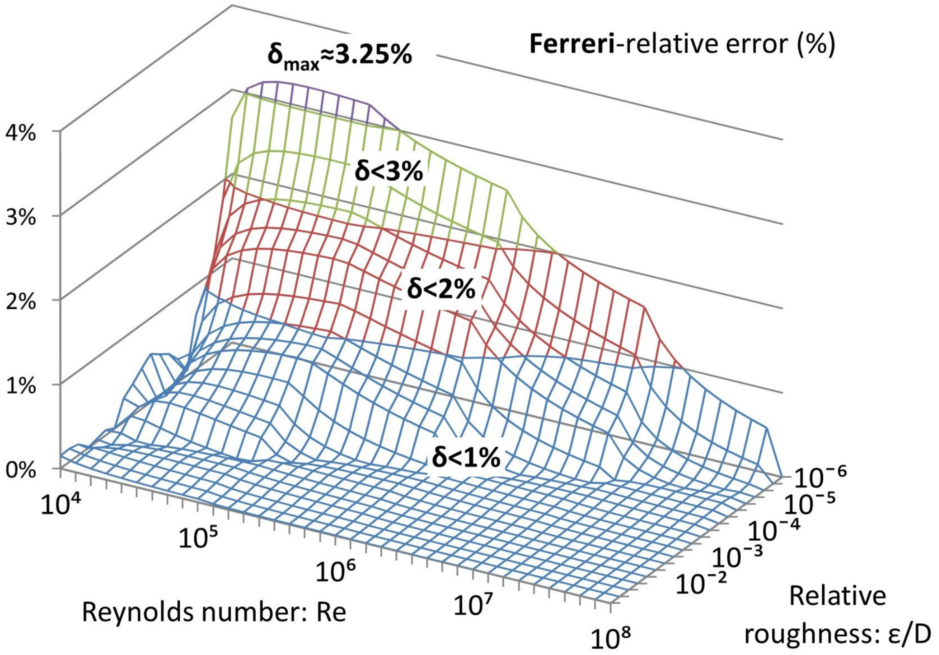

Figure A1. Distribution of the relative error δ% of the simpler version of Ferreri's explicit approximation from Equation (A1), with the maximum relative error δ%$_{max}$ = 3.25%.

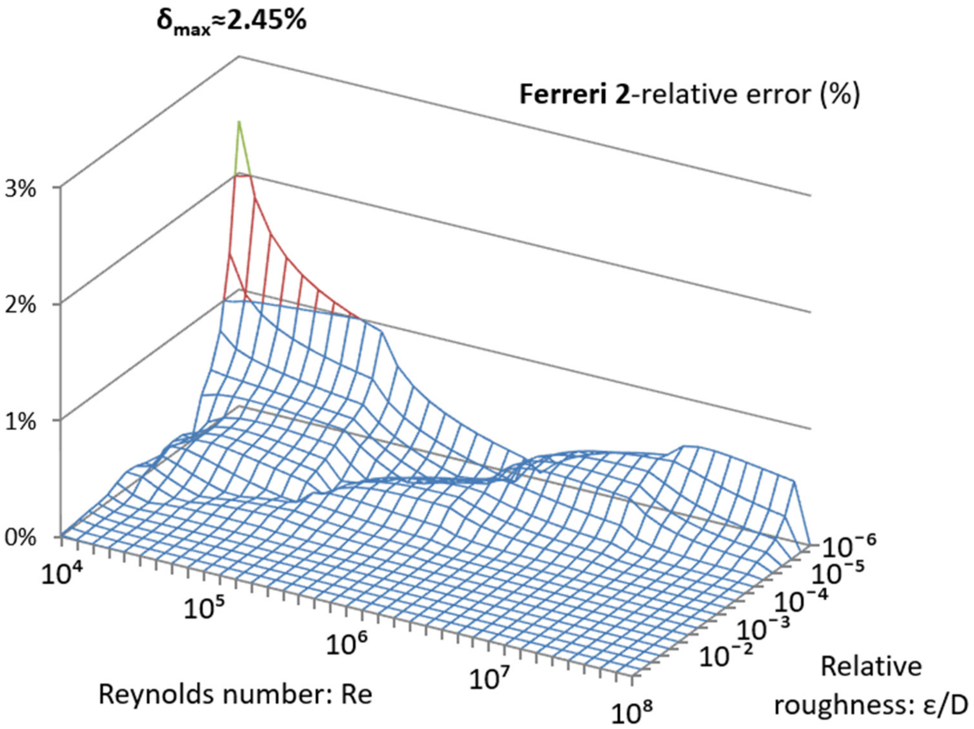

Figure A2. Distribution of the relative error δ% of the more complex version of Ferreri's explicit approximation from Equation (A2), with the maximum relative error $\delta\%_{max}$ = 2.45%.

Appendix C.2. Top-Performance Approximations

Numerous explicit approximations introduce the maximum relative error $\delta\%_{max} < 0.1\%$. They were developed by Praks and Brkić [16], Serghides [17], Vatankhah [18], Romeo et al. [19], Buzzelli [20], Offor and Alabi [21], and Lamri [22,23]. Approximations can be compared not only by accuracy but also by their computational efficiency. Winning and Coole [24] estimated the execution speed of mathematical operations and functions, which can give an overall estimation of how fast each approximation is. For the sake of comparison, according to [15], the top approximations according to these two criteria are as follows:

- The most accurate approximation was made by Praks and Brkić [16], Equation (A3), with $\delta\%_{max} < 0.001204\%$, and its execution time for the same task is about 450.7 s,
- The Lamri approximation [22], Equation (A4), with $\delta\%_{max} < 0.097438\%$, requires about 440.2 s and is, to date, the fastest explicit approximation available for computer execution.

$$\left.\begin{array}{c}\frac{1}{\sqrt{\lambda}} \approx 0.8685972\cdot(B+y)\\ A \approx \frac{Re\cdot\varepsilon}{8.0897}\\ B \approx ln(Re)-0.779626\\ x \approx A+B\\ C \approx ln(x)\\ y \approx \frac{C}{x-0.5588\cdot C+1.2079}-C\end{array}\right\}, \tag{A3}$$

$$\left.\begin{array}{c}\frac{1}{\sqrt{\lambda}} \approx \mathrm{A}_1+0.8686\cdot\left(\frac{0.8686}{\mathrm{B}_1}-1\right)\cdot\ln(\mathrm{B}_1)\\ \mathrm{A}_1 \approx 0.8686\cdot\ln\left(\frac{\mathrm{Re}}{2.51}\right)\\ \mathrm{B}_1 \approx \mathrm{A}_1+\frac{\mathrm{Re}\cdot\varepsilon}{9.3125}\end{array}\right\}, \tag{A4}$$

Something similar to λ_∞ was created for the fully developed rough turbulent regime defined in Equation (2), $A_1 = \frac{1}{\sqrt{\lambda_{smooth}}}$, where λ_{smooth} is the dimensionless Darcy friction factor valid for hydraulically smooth pipes.

The accuracy of Equations (A3) and (A4) was also evaluated using the methodology and datasets described in this Data Descriptor.

References

1. Colebrook, C.F. Turbulent flow in pipes, with particular reference to the transition region between the smooth and the rough pipe laws. *J. Inst. Civ. Eng.* **1939**, *11*, 133–156. [CrossRef]
2. Colebrook, C.F.; White, C. Experiments with fluid friction in roughened pipes. *Proc. R. Soc. London. Ser. A Math. Phys. Sci.* **1937**, *161*, 367–381. [CrossRef]
3. Brkić, D.; Praks, P. Accurate and efficient explicit approximations of the Colebrook flow friction equation based on the Wright ω-function. *Mathematics* **2019**, *7*, 34. [CrossRef]
4. Hayes, B. Why W? *Am. Sci.* **2005**, *93*, 104–108. [CrossRef]
5. Sonnad, J.R.; Goudar, C.T. Constraints for using Lambert W function-based explicit Colebrook–White equation. *J. Hydraul. Eng.* **2004**, *130*, 929–931. [CrossRef]
6. Rollmann, P.; Spindler, K. Explicit representation of the implicit Colebrook–White equation. *Case Stud. Therm. Eng.* **2015**, *5*, 41–47. [CrossRef]
7. Biberg, D. Fast and accurate approximations for the Colebrook equation. *J. Fluids Eng.* **2017**, *139*, 031401. [CrossRef]
8. 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]
9. Brkić, D. Review of explicit approximations to the Colebrook relation for flow friction. *J. Pet. Sci. Eng.* **2011**, *77*, 34–48. [CrossRef]
10. Zigrang, D.J.; Sylvester, N.D. A review of explicit friction factor equations. *J. Energy Resour. Technol.* **1985**, *107*, 280–283. [CrossRef]
11. Gregory, G.A.; Fogarasi, M. Alternate to standard friction factor equation. *Oil Gas. J.* **1985**, *83*, 120–127.
12. Ferreri, G.B. Explicit approximation of the Colebrook-White formula based on the friction Reynolds number. *Eur. J. Mech. B/Fluids* **2025**, *114*, 204349. [CrossRef]
13. Ferreri, G.B. A new approach for explicit approximation of the Colebrook–White formula for pipe flows. *J. Hydroinformatics* **2024**, *26*, 1558–1571. [CrossRef]
14. Zigrang, D.J.; Sylvester, N.D. Explicit approximations to the solution of Colebrook's friction factor equation. *AIChE J.* **1982**, *28*, 514–515. [CrossRef]
15. Brkić, D.; Stajić, Z. Excel VBA-Based User Defined Functions for highly precise Colebrook's pipe flow friction approximations: A comparative overview. *Facta Univ. Ser. Mech. Eng.* **2021**, *19*, 253–269. [CrossRef]
16. Praks, P.; Brkić, D. Review of new flow friction equations: Constructing Colebrook's explicit correlations accurately. *Rev. Int. Métodos Numéricos Cálculo Diseño Ing.* **2020**, *36*, 41. [CrossRef]
17. Serghides, T.K. Estimate friction factor accurately. *Chem. Eng.* **1984**, *91*, 63–64.
18. Vatankhah, A.R. Approximate analytical solutions for the Colebrook equation. *J. Hydraul. Eng.* **2018**, *144*, 06018007. [CrossRef]
19. Romeo, E.; Royo, C.; Monzón, A. Improved explicit equations for estimation of the friction factor in rough and smooth pipes. *Chem. Eng. J.* **2002**, *86*, 369–374. [CrossRef]
20. Buzzelli, D. Calculating friction in one step. *Mach. Des.* **2008**, *80*, 54–55. Available online: https://www.machinedesign.com/archive/article/21817480/calculating-friction-in-one-step (accessed on 1 November 2025).
21. Offor, U.H.; Alabi, S.B. An accurate and computationally efficient explicit friction factor model. *Adv. Chem. Eng. Sci.* **2016**, *6*, 237–245. [CrossRef]
22. Lamri, A.A. Discussion of "Approximate analytical solutions for the Colebrook equation". *J. Hydraul. Eng.* **2020**, *146*, 07019012. [CrossRef]
23. Lamri, A.A.; Easa, S.M. Computationally efficient and accurate solution for Colebrook equation based on Lagrange theorem. *J. Fluids Eng.* **2022**, *144*, 014504. [CrossRef]
24. 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]
25. Brkić, D. Solution of the implicit Colebrook equation for flow friction using Excel. *Spreadsheets Educ.* **2017**, *10*, 4663. Available online: https://sie.scholasticahq.com/article/4663.pdf (accessed on 1 November 2025).
26. Zhao, Q.; Wu, W.; Simpson, A.R.; Willis, A. Simpler Is Better—Calibration of Pipe Roughness in Water Distribution Systems. *Water* **2022**, *14*, 3276. [CrossRef]
27. Muzzo, L.E.; Matoba, G.K.; Ribeiro, L.F. Uncertainty of pipe flow friction factor equations. *Mech. Res. Commun.* **2021**, *116*, 103764. [CrossRef]

28. Moody, L.F. Friction factors for pipe flow. *Trans. ASME* **1944**, *66*, 671–684. [CrossRef]
29. Brkić, D. Revised Friction Groups for Evaluating Hydraulic Parameters: Pressure Drop, Flow, and Diameter Estimation. *J. Mar. Sci. Eng.* **2024**, *12*, 1663. [CrossRef]
30. Mishra, R.; Ojha, C.S.P. Application of AI-based techniques on Moody's diagram for predicting friction factor in pipe flow. *J.* **2023**, *6*, 544–563. [CrossRef]
31. LaViolette, M. On the history, science, and technology included in the Moody diagram. *J. Fluids Eng.* **2017**, *139*, 030801. [CrossRef]
32. de Souza Mendes, P.R. A Note on the Moody Diagram. *Fluids* **2024**, *9*, 98. [CrossRef]
33. Huang, S. Reading the Moody chart with a linear interpolation method. *Sci. Rep.* **2022**, *12*, 6587. [CrossRef]
34. Yıldırım, G. Computer-based analysis of explicit approximations to the implicit Colebrook–White equation in turbulent flow friction factor calculation. *Adv. Eng. Softw.* **2009**, *40*, 1183–1190. [CrossRef]
35. Shaikh, M.M.; Wagan, A.I. A sixteen decimal places' accurate Darcy friction factor database using non-linear Colebrook's equation with a million nodes: A way forward to the soft computing techniques. *Data Brief.* **2019**, *27*, 104733. [CrossRef]
36. Praks, P.; Brkić, D. Approximate Flow Friction Factor: Estimation of the Accuracy Using Sobol's Quasi-Random Sampling. *Axioms* **2022**, *11*, 36. [CrossRef]
37. Chi, H.; Mascagni, M.; Warnock, T. On the optimal Halton sequence. *Math. Comput. Simul.* **2005**, *70*, 9–21. [CrossRef]
38. Faure, H.; Lemieux, C. Generalized Halton sequences in 2008: A comparative study. *ACM Trans. Model. Comput. Simul.* **2009**, *19*, 1–31. [CrossRef]
39. Wang, X.; Hickernell, F.J. Randomized Halton Sequences. *Math. Comput. Model.* **2000**, *32*, 887–899. [CrossRef]
40. Wong, T.T.; Luk, W.S.; Heng, P.A. Sampling with Hammersley and Halton points. *J. Graph. Tools* **1997**, *2*, 9–24. [CrossRef]
41. Chen, H.; Yang, C.; Deng, K.; Zhou, N.; Wu, H. Multi-objective optimization of the hybrid wind/solar/fuel cell distributed generation system using Hammersley Sequence Sampling. *Int. J. Hydrogen Energy* **2017**, *42*, 7836–7846. [CrossRef]
42. Sobol, I.M. Uniformly distributed sequences with an additional uniform property. *USSR Comput. Math. Math. Phys.* **1976**, *16*, 236–242. [CrossRef]
43. Weirs, V.G.; Kamm, J.R.; Swiler, L.P.; Tarantola, S.; Ratto, M.; Adams, B.M.; Rider, W.J.; Eldred, M.S. Sensitivity analysis techniques applied to a system of hyperbolic conservation laws. *Reliab. Eng. Syst. Saf.* **2012**, *107*, 157–170. [CrossRef]
44. Zhang, F.; Tian, Y.; Liu, Q.; Gao, Y.; Wang, X.; Liu, Z. Uncertainty Analysis of Performance Parameters of a Hybrid Thermoelectric Generator Based on Sobol Sequence Sampling. *Appl. Sci.* **2025**, *15*, 9180. [CrossRef]
45. Kannan, S.K.; Diwekar, U. An Enhanced Particle Swarm Optimization (PSO) Algorithm Employing Quasi-Random Numbers. *Algorithms* **2024**, *17*, 195. [CrossRef]
46. Parreira, T.G.; Rodrigues, D.C.; Oliveira, M.C.; Sakharova, N.A.; Prates, P.A.; Pereira, A.F.G. Sensitivity Analysis of the Square Cup Forming Process Using PAWN and Sobol Indices. *Metals* **2024**, *14*, 432. [CrossRef]
47. Zheng, W.; Ai, Y.; Zhang, W. Improved Snake Optimizer Using Sobol Sequential Nonlinear Factors and Different Learning Strategies and Its Applications. *Mathematics* **2024**, *12*, 1708. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.