

## Systematic Review

# Physics-Informed Neural Network (PINN) Evolution and Beyond: A Systematic Literature Review and Bibliometric Analysis

Zaharaddeen Karami Lawal<sup>1</sup>, Hayati Yassin<sup>1,2,\*</sup>, Daphne Teck Ching Lai<sup>2,3</sup>  and Azam Che Idris<sup>1</sup><sup>1</sup> Faculty of Integrated Technologies, Universiti Brunei Darussalam, Gadong BE1410, Brunei<sup>2</sup> Institute of Applied Data Analytics (IADA), Universiti Brunei Darussalam, Gadong BE1410, Brunei<sup>3</sup> School of Digital Science, Universiti Brunei Darussalam, Gadong BE1410, Brunei

\* Correspondence: hayati.yassin@ubd.edu.bn

**Abstract:** This research aims to study and assess state-of-the-art physics-informed neural networks (PINNs) from different researchers' perspectives. The PRISMA framework was used for a systematic literature review, and 120 research articles from the computational sciences and engineering domain were specifically classified through a well-defined keyword search in Scopus and Web of Science databases. Through bibliometric analyses, we have identified journal sources with the most publications, authors with high citations, and countries with many publications on PINNs. Some newly improved techniques developed to enhance PINN performance and reduce high training costs and slowness, among other limitations, have been highlighted. Different approaches have been introduced to overcome the limitations of PINNs. In this review, we categorized the newly proposed PINN methods into Extended PINNs, Hybrid PINNs, and Minimized Loss techniques. Various potential future research directions are outlined based on the limitations of the proposed solutions.

**Keywords:** physics-informed neural networks (PINNs); partial differential equation (PDE); loss function; activation function; deep learning



**Citation:** Lawal, Z.K.; Yassin, H.; Lai, D.T.C.; Che Idris, A. Physics-Informed Neural Network (PINN) Evolution and Beyond: A Systematic Literature Review and Bibliometric Analysis. *Big Data Cogn. Comput.* **2022**, *6*, 140. <https://doi.org/10.3390/bdcc6040140>

Academic Editors: Jenq-Haur Wang and Min Chen

Received: 31 August 2022

Accepted: 15 November 2022

Published: 21 November 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Physics-informed neural networks (PINNs) [1] are frequently employed to address a variety of scientific computer problems. Due to their superior approximation and generalization capabilities, physics-informed neural networks have gained popularity in solving high-dimensional partial differential equations (PDEs) [2]. As a deep learning method, physics-informed neural networks bridge the gap between machine learning and scientific computing. PINNs have contributed to improvements in many areas of computer science and engineering due to their simplicity [3,4]. In the engineering and scientific literature, PINNs are receiving more attention for solving a variety of differential equations with applications in weather modeling, healthcare, manufacturing, and other fields [5–7]. However, PINNs are not suitable for several real-time applications because of their high training costs. Although various proposals have been made to enhance the training effectiveness of PINNs, only some have considered the effects of initialization [8–10]. Another obstacle to the application of PINNs to a wide range of real-world problems is their poor scalability [5].

Due to the sheer number of residual points in time-constrained space, an accurate network, called a physics-informed neural network (PINN), can be trained by minimizing only the residual loss. Moreover, the prediction of the high-fidelity solution for complex nonlinear problems in low-dimensional space is more accurate than the solutions of the reduced-order equations [1,11–13].

The ability of PINNs to learn from sparse input is one of their most well-known features [12,14,15]. Initial and boundary terms are not systematically given in the context of PINN structure; therefore, they need to be incorporated in the loss function of the network,

which must be learned simultaneously with the uncertain functions of the differential equation (DE) [16,17]. While implementing gradient-based approaches, combining multiple targets during the network's training can lead to biased gradients, leading to PINNs failing to appropriately learn the fundamental DE solution [18–20].

According to Raissi et al. [1], Schiassi et al. [21], and Zhang et al. [22], the key drawback of conventional PINNs is that even the DE limitations are still not mathematically solved, hence they need be learned concurrently with the DE solutions within the domain. As a result, we cope with competing goals during PINN training: learning the concealed DE solutions well in domains while also learning the hidden DE solutions on the boundary [21,23,24]. This results in imbalanced gradients during network training using gradient-based approaches, causing PINNs to fail to learn the basic DE solutions accurately [21,25]. According to Dwivedi et al. [26], despite the numerous benefits that PINNs provide, they have three major drawbacks. The first is their slowness [26,27] when applied to real problems; PINNs use up gradient descent optimization and are quite slow when compared to other numerical approaches. For highly deep networks, PINNs are vulnerable to vanishing gradient problems [6,26,28]. There is also the possibility that a solution will become stuck at a minimal point. Finally, the PINN's learning process is fine-tuned by hand. We cannot ascertain exactly how much data or even which framework is sufficient for a particular set of sample instances [1,29–31].

The weighted least-squares collocation approach utilized in PINNs can be interpreted as a hybrid physics/data loss scheme [32–34]. As a result, PINNs have inherited several drawbacks common to such approaches, including the necessity to evaluate PDE residuals correctly against beginning as well as boundary conditions; a severe regularity demand for solutions remains continuous, as does the inability of natural methods to impose conservation structures [32,35–37].

Although PINNs have been exceptionally beneficial to the scientific community, Colby et al. [18] discovered that they are often incapable of appropriately solving a mathematical model for interfacial problems used for solidification dynamics called “phase field problems”. As a result, they found that specific elements of phase field model solutions (both spatially and dynamically) were more difficult to learn than others. These problematic places may alter as you discover the solution.

The goal of this research is to find physics-informed neural network adaptations for solving various problems from the literature and to highlight newly improved PINN methodologies that have been proposed using different techniques. The main objectives of the study were to evaluate the current state of the art in this field of research using numerous bibliometric analyses, identify the full spectrum of eligibility requirements studied in the literature through information synthesis, create a collection of general information about how far research on PINNs has changed over time, and identify newly introduced PINN approaches while highlighting some topics for future research. We tried to categorize state-of-the-art PINN techniques into three groups. These are Extended, Hybrid, and Minimized Loss PINN techniques and will be discussed later in Section 4.1. In this literature review, we aimed to answer the following question: *What techniques have been introduced to optimize the performance of physics-informed neural networks?* Although PINNs are utilized to solve problems in practically all the domains of human endeavor, throughout this review, we focused on computational sciences and engineering.

The rest of the paper is organized as follows: Section 2 presents the background. Section 3 explores the quality assessment and qualitative synthesis used in the literature review for this study. Section 4 discusses the results of the bibliometric analysis, as well as the objectives, methods, and limitations of the newly proposed PINN techniques. Sections 5 and 6 discuss future research directions and conclusions, respectively.

## 2. Background

Many methods for solving differential equations have been established over the years. Some generate a solution in an array containing the solution's value in a predefined set

of locations. Others employ basis-functions to designate the solution in analytic methods and typically translate the genuine problem to a system of algebraic equations. Most past projects seeking to solve partial differential equations with neural networks have been limited to the case of solving methods of algebraic equations that come from domain discretization. In 1998, Isaac Lagaris et al. [38] introduced an improved method for solving ordinary DEs and partial differential equations by employing Artificial Neural Networks (ANNs). In their newly introduced method, the differential equation's trial solution was represented by the sum of two parts. There were no adjustable parameters in the first half, which satisfied the initial/boundary conditions. The second component was designed in a way that did not change the initial or boundary conditions. In this section, a feedforward neural network (NN) with programmable constraints (the weights) was used. Therefore, the network is trained to fit the differential equations together with the initial/boundary conditions that were satisfied by construction [38] methods and can be used for a range of ordinary differential equations (ODEs), coupled ODE systems, and partial differential equations (PDEs).

In 2011, Ladislav Zjavka [39] developed a new technique, known as a Differential Polynomial Neural Network (D-PNN). The proposed D-PNN technique approximates a multi-parametric function by generating and solving unknown partial DEs. A differential equation is substituted to create a system model of dependent variables, leading to the summation of fractional polynomial derivative terms. In contrast to the ANN method, the D-PNN allows each neuron to directly participate in the calculation of the network's overall output. Consequently, in 2013, Ladislav et al. [40] showed that D-PNNs could be applied to solve complex mathematical problems.

In 2015, Ladislav et al. [41] presented a recurrent neural network (RNN) of one layer, which is frequently employed for time series predictions and which was used for comparison. With twenty-four succession samples constantly generated by the benchmark and a continuous step value of 0.1, in the range of 0–2.4, the D-PNN and RNN were trained. Two incredibly different networks were trained with only a relatively narrow range of values, which did not accurately reflect function-specific progress over an entire period. The models were then tested over a longer period. The networks only estimated one benchmark value for the next step  $x$  using the calculated accurate function  $f(x)$  with three-input sequences, rather than making entire predictions built on prior steps' outputs (approximations).

In 2017, Raissi et al. [42] proposed hidden physics models (machine learning of nonlinear partial DEs). To obtain patterns from the high-dimensional data produced by experiments, the models are essentially data-efficient learning approaches that can exploit underlying physical laws expressed by time dependency and nonlinear PDEs. The proposed methodology can be used to solve learning problems, identify systems, or find partial differential equations using data. Within the same time frame, Raissi, Perdikaris, and Karniadakis [43] introduced the prototype of novel physics-informed neural networks (PINNs). It was referred to as "data-driven solutions of nonlinear partial differential equations". The term "physics informed neural networks" refers to NNs that have been trained to solve any supervised learning tasks while complying by any identifiable physics law as expressed by general nonlinear PDEs. They developed two unique kinds of algorithms, namely continuous time and discrete time models, depending on the type and organization of the available data. The result is a new family of universal function approximators that efficiently use data and intuitively encode any underlying physical constraints as prior knowledge.

In 2018, Raissi et al., proposed another technique called "multistep neural networks for data-driven discovery of nonlinear dynamical systems" [44]. They presented a machine learning strategy for recognizing nonlinear dynamical systems using data. To clarify the principles governing the evolution of a particular dataset, they specifically combined traditional numerical analysis methods, such as multi-step time-stepping patterns, with potent nonlinear function approximators, such as deep neural networks. They described how this allowed them to accurately learn dynamics, predict future states, and identify

points of attraction. They evaluated their methods for a set of benchmark problems that required the identification of complicated, nonlinear, and chaotic dynamics.

In the same period, Raissi [45] proposed a new technique called Deep-Hidden Physics Models to support the use of deep learning in the discovery of nonlinear PDEs. To find nonlinear PDEs using dispersed and possibly noisy observations in space and time, they proposed a deep learning approach. They used two different deep neural networks to estimate the unknown solution and the nonlinear dynamics. They were able to prevent numerical differentiations because they were intrinsically unstable and ill-conditioned by acting as a prior on the unknown answer. The second network exhibited nonlinear dynamics and aids in our ability to identify the fundamental principles guiding the evolution of a certain spatiotemporal dataset. They demonstrated how the developed model could help them to grasp the structure and dynamics of a system and forecast its future state by testing the efficiency of the approach on a variety of benchmark problems covering a range of scientific areas.

The following are various adjustments to the basic PINN prototype. In early 2019, Raissi et al. [1] presented the full version of PINNs as a “Deep Learning Framework for Solving Forward and Inverse Problems Involving Nonlinear Partial Differential Equations”.

Despite the significant improvement in simulating multi-physics problems by using numerical modeling of PDEs, noisy data cannot be totally integrated into current algorithms. Mesh creation is still complex and challenging and high-dimensional problems constrained by parameterized PDEs cannot be solved [46–48]. Furthermore, inverse problems that involve hidden physics are often extremely expensive to solve. This is because they require complex computer code and several formulations [49–51].

In the recent past, deep learning and physics-informed neural networks (PINNs) have garnered a lot of interest in scientific research and engineering applications [52–54]. Deep learning has lately evolved as a modern discipline of scientific machine learning in relation to solving the governing PDEs of physical situations [55–57] by incorporating the universal approximation and high expressivity of neural networks [58–60]. Deep neural networks can estimate any high-dimensional function if adequate training data are available [61–63].

However, these networks do not consider the physical properties of the problem, and the accuracy of the approximation they provide still depends heavily on the exact geometry of the problem and the initial and boundary conditions [64–66]. The solution is indeed not unique without this basic information, and it may lose physical accuracy. Data are scarce and imprecise in most applications, and the controlling physics are unknown, limiting the utility of traditional machine learning (ML) and physics-based approaches [56,67–70].

Physics-informed neural networks, on the contrary, use fundamental physical equations to train neural networks. PINNs should be trained to comply with both the available training data and the enforced governing equations. A neural network could be managed in this way by training data that need not be huge and full [58,61,71].

Large volumes of data are required in a wide range of applications to train the neural network through decreasing the distance (loss) in between network output and the ground truth [72–75].

A PINN’s loss function is made up of numerous terms, such as governing equations and boundary conditions. To enable a network model with well-known physical laws, Raissi et al. [1] proposed physics-informed neural networks (PINNs), which were intended to resolve direct as well as inverse problems controlled by numerous different types of PDEs [1,72,76].

In partial and ordinary differential equations, PINNs were applied to learn solutions and parameters. The basic concept, especially with physics-informed neural networks, is to employ physics laws in the form of DEs to train NNs [77]. This is significantly distinctive from using the neural networks as dummy models trained with data from a composition of input and output values for nonlinear PDEs [1,14,78].

The goal is to train neural networks not only on data, as is common in deep learning, but also on the fundamental model of the DEs [37,79]. Without understanding the boundary

conditions, it is feasible to get a precise solution to PDEs [34,62,80]. Therefore, PINNs can be applied to identify an optimum solution of high reliability with some insight on the physical properties of the problems and training data (even sparse and partial).

In other words, PINNs provide an inconsistent method for solving differential equations that violates any mathematical discretization of the system [37,81,82]. PINNs show improvement in terms of performance and accuracy while only using a small amount of data for training. They are used to perfectly describe the physical attributes of a system's dynamic ecosystem [83–85].

PINNs provide the solutions to a broad range of computational science problems and are a pioneering technology that is leading towards the advancement of new categories of numerical solvers for PDEs. PINNs can be viewed as a meshless alternative to classic methodologies (for example, CFD for fluid dynamics) and novel data-driven methods for model inversion and system identification [1,3,86].

Moreover, the trained PINNs could predict values on modeled grids of various resolutions without having to be retrained [87]. In addition, they allow for the manipulation of automatic differentiation (AD) [88,89] when calculating the appropriate derivatives in PDEs, a novel class of differentiation techniques frequently utilized to derive NNs that have been shown to be exceptional regarding numeric or symbolic differentiation [87,88,90].

### 2.1. Physics-Informed Neural Networks

PDEs with high dimensions are commonly used in a variety of disciplines, including physics, chemistry, engineering, finance, and more [91]. Higher dimensions make numerical PDE computational methods such as finite difference or finite element methods impractical due to the explosion in the number of grid points and the need for smaller time steps.

Physics-informed neural networks (PINNs) are models developed to obey physical laws specified by (nonlinear) partial differential equations (PDEs). They can be used for supervised tasks where the reduction of errors with respect to data and physical laws is required [1,91,92].

The loss function is defined as follows:

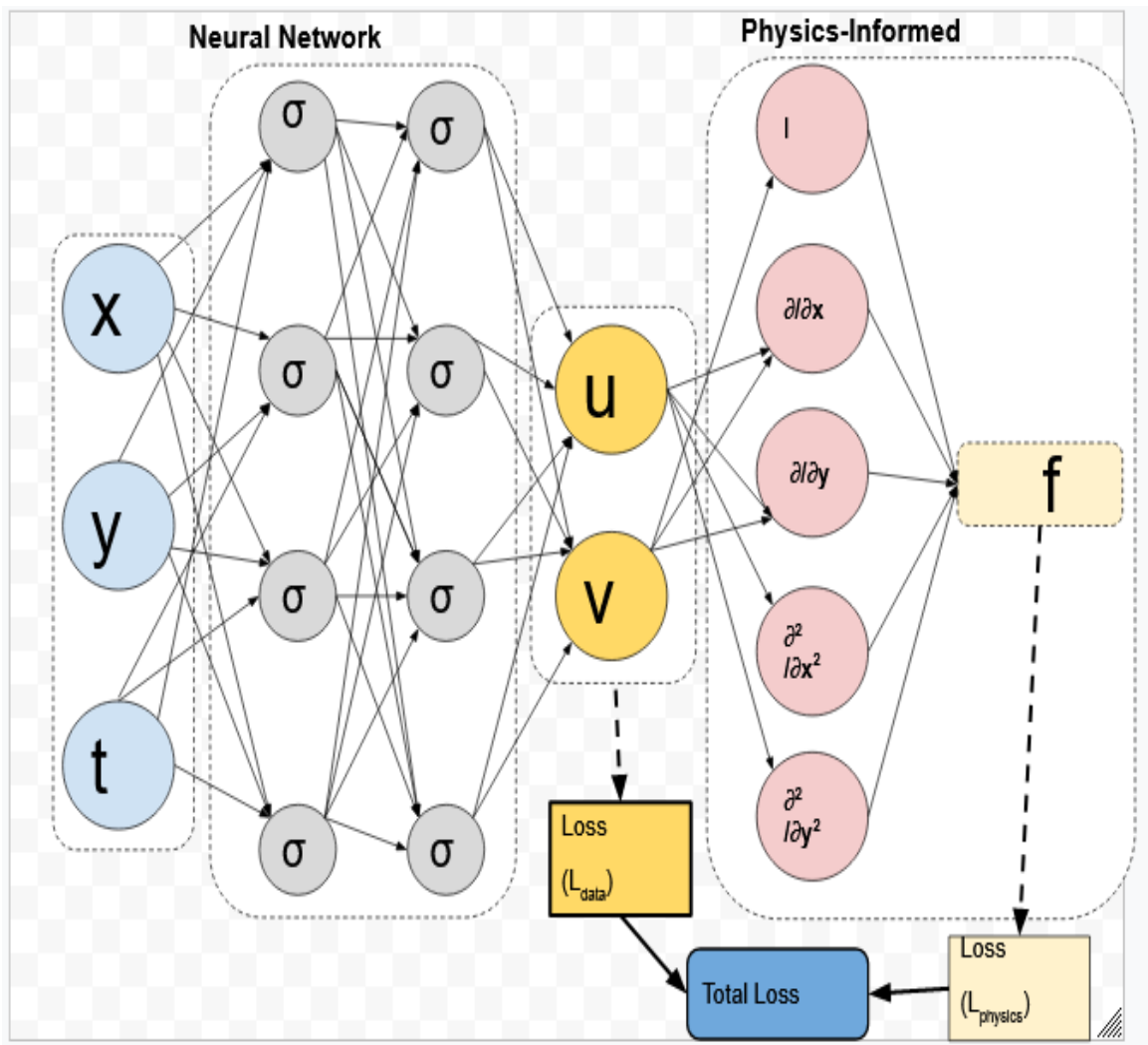
$$l = l_{data} + w l_{physics} \quad (1)$$

where  $l_{data}$  is the loss with respect to the data,  $w l_{physics}$  is a physics law's loss relative to the PDEs, and  $w$  is the weight matrix of a physics loss. By using mean-squared error for both losses, if  $f_i$  is the error of each physical state, we derive:

$$l = \sum_i (y_i - \hat{y})^2 - w \sum_i f_i^2 \quad (2)$$

The input values  $x$ ,  $y$ , and  $t$  are used in the architecture of the network, and the output values are  $u$  and  $v$ . The output variables are used directly to calculate the data loss term. However, we distinguish the variables with respect to the input variables for the physical loss term to account for the physical loss function, as shown in Figure 1 [91] below. The goal is to minimize deviations from the physics law by monitoring training with measured or generated data (i.e.,  $l_{data}$ ), as well as by training to diminish the departure from the physics law. To further train to minimize deviation from the physical law, by preventing overfitting, the term  $l_{physics}$  ensures that such a neural network generalizes better given unknown inputs. The physical loss allows the output variables to be trained to include their first and/or second-degree derivatives, as well as a local region around each input value of the given data (depending on the PDEs).





**Figure 1.** Schematic diagram of a PINN. Adapted with permission from Ref. [91]. Copyright 2021 De Wolff, Carrillo, Martí, and Sanchez-Pi.

## 2.2. Modeling and Computation

A general nonlinear partial differential equation can be:

$$\frac{\partial u}{\partial t} = 0$$

$$u_t + N[u; \lambda] = 0, \quad x \in \Omega, \quad t \in [0, T] \quad (3)$$

$u(t, x)$  denotes the solution,  $N[u; \lambda]$  is a nonlinear operator parametrized by  $\lambda$ , and  $\Omega$  is a subset of  $\mathbb{R}^D$ .

Numerous problems in mathematical physics, such as conservative laws, diffusion processes, advection–diffusion systems, and kinetic equations, fall under this general category of governing equations.

PINNs can be programmed to solve two major types of PDE problems on the noisy data of a general dynamical system denoted by the above equation [14,92,93]. These are:

1. Data-driven solutions.
2. Data-driven discovery.
- Data-Driven solutions of Partial Differential Equations

The data-driven solution of PDE findings when calculating the unknown state  $u(t, x)$  of the system given noisy measurements  $z$  of the state and fixed model parameters  $\lambda$  reads as follows:

$$u_t + N[u] = 0, \quad x \in \Omega, \quad t \in [0, T] \quad (4)$$

By defining  $f(t, x)$  as

$$f := u_t + N[u] = 0 \quad (5)$$

and approximating  $u(t, x)$  with a deep neural network,  $f(t, x)$  results in a PINN. This network can be differentiated using automatic differentiation. The parameters of  $u(t, x)$  and  $f(t, x)$  can be then learned by minimizing the following loss function  $L_{tot}$ :

$$L_{tot} = L_u + L_f \quad (6)$$

where  $L_u = ||u - z||_T$  with  $u$  and  $z$  representing state solutions and measurements at sparse location  $T$ , respectively, and  $L_f = ||f||_T$  is a residual function. To satisfy this second term, the structured data defined by the PDEs must be used during the training phase [81].

This approach enables the creation of physically informed surrogate models that are computationally efficient and which can be used for simulations, model predictive control, and data-driven physical process predictions [1,94–96].

- Data Discovery of Partial Differential Equations

The data-driven discovery of PDEs leads to computation of the unknown states  $u(t, x)$  and  $f(t, x)$  of the system and the learning of model parameters,  $\lambda$ , that most accurately capture the observed data:

$$u_t + N[u; \lambda] = 0, \quad x \in \Omega, \quad t \in [0, T] \quad (7)$$

Additionally,  $f(t, x)$  is defined as

$$f := u_t + N[u; \lambda] = 0 \quad (8)$$

Further, approximating  $u(t, x)$  with a deep neural network,  $(t, x)$ , leads to a PINN. This network can be derived by automatic differentiation. The parameters of  $u(t, x)$  and  $f(t, x)$  can then be learned along with the parameters,  $\lambda$ , of the differential operator by minimizing the following loss function  $L_{tot}$ :

$$L_{tot} = L_u + L_f \quad (9)$$

where  $L_u = ||u - z||_T$  with  $u$  and  $z$  representing state solutions and measurements at sparse location  $T$ , respectively, and  $L_f = ||f||_T$  is a residual function. This second term involves the well-defined information characterized by the PDEs to be satisfied in the training procedure.

### 3. Methodology

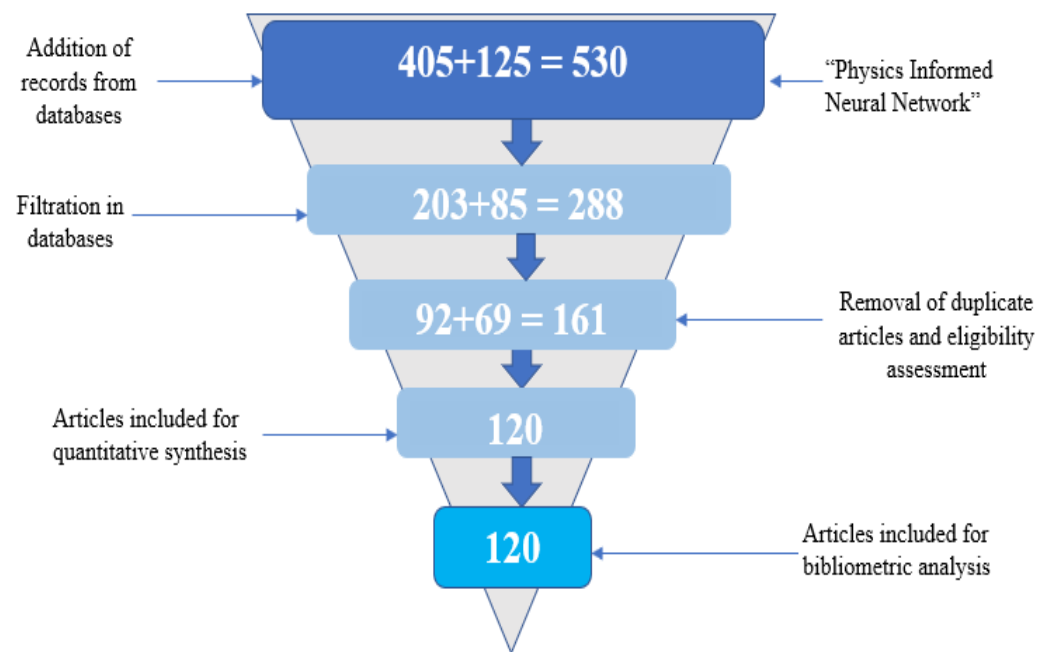
For reviewing current research, this paper used the PRISMA framework [97–99]. The scoping approach was utilized to retrieve the most relevant papers on physics-informed neural networks. This method aided the control of critical mandatory components and the classification of potential search terms [97,100,101]. To identify relevant scientific papers and articles, multiple databases were searched. A search using a single keyword (“physics-informed neural networks”) was conducted to find relevant publications from the most reputable and reliable research resources. Scopus and Web of Science were used, along with the Web of Science core collection, Derwent Innovations Index, MEDLINE, KCI-Korean Journal Database, and SCIELO Citation Index.

The keyword “Physics Informed Neural Network” was solely used in each database search for the relevant literature. Predefined inclusion and exclusion criteria and quality

requirements were used to refine the data search. Each filter verified that the quality requirement was met, and the next section discusses the inclusion and exclusion criteria.

Because our search query was put in a double quote, we employed deterministic information retrieval to look for suitable papers, as we described earlier. The literature searches in all the databases listed above retrieved articles from 2019 to 2022. Initially, 530 items were found; however, this was largely made up of a variety of materials, such as research articles, reviews, editorials, and book chapters, among others.

Many researchers use PINNs to solve problems in different areas of human endeavor. We have limited our research to computational sciences and engineering and focused on research articles, review papers, and book chapters in our literature search. A total of 288 documents were chosen, as illustrated in Figure 2. Articles from computational sciences and engineering were chosen as the following sequence. This included computer science, engineering, mathematics, and physics. Non-English documents were also excluded. The PRISMA checklist can be found in Supplementary Materials.



**Figure 2.** PRISMA framework.

Duplicate articles were also removed from the Scopus and Web of Science metadata files. The two files were combined and duplicate journal articles removed. In this review, the PRISMA 2020 flow diagram was used, as shown in Figure 3 below.



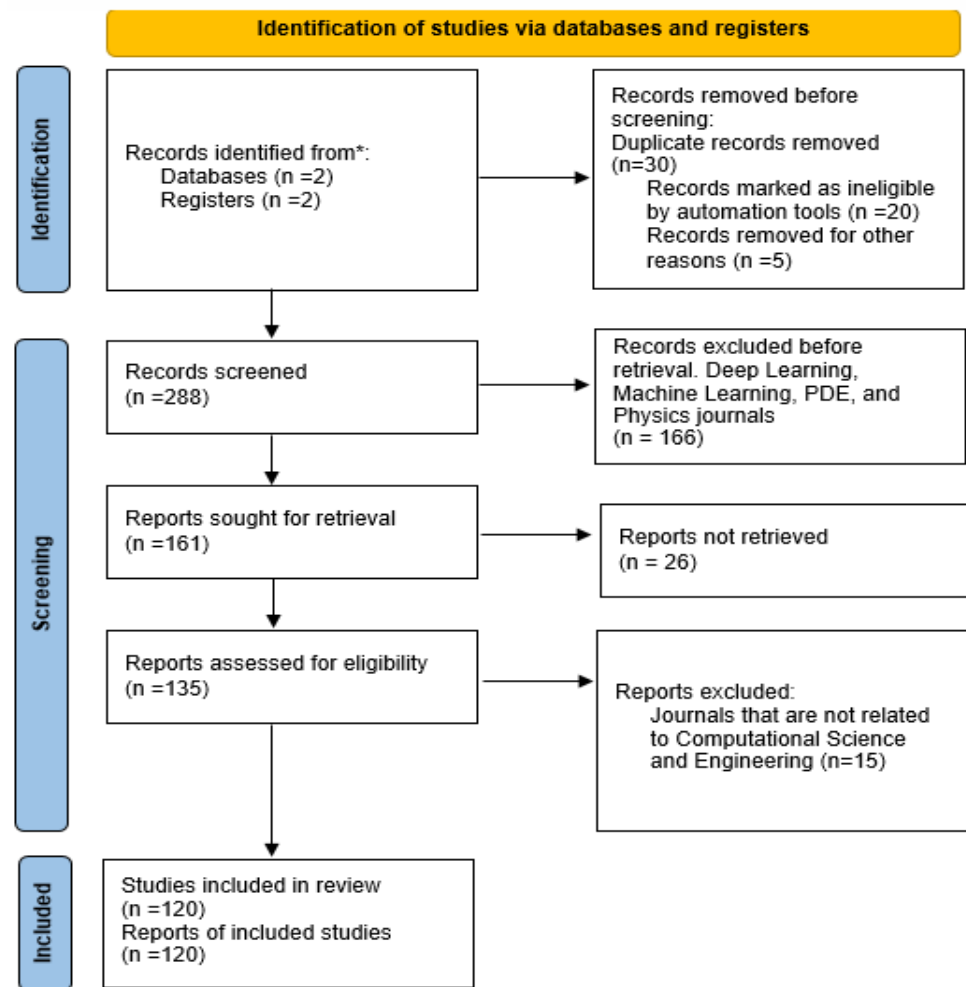


Figure 3. PRISMA flow diagram.

### 3.1. Quality Assessment

This review looks at final published journal articles, reviews, and conference papers to find the best results and capture an excellent overview of previous data. Abstracts and conclusions were separated to keep the archive to a minimum. In addition, cited references in the evaluated articles were considered. As stated earlier, the two metafile records were combined. The duplicate records were eliminated to improve the findings. Irrelevant data were also excluded.

### 3.2. Qualitative Synthesis Used in the Literature Review

After selecting the documents, a two-step approach was used to confirm the quality of the analysis performed on the published papers. The relevant metadata were initially imported into Microsoft Excel to conduct a descriptive study of physics-informed neural network literature, which included identification of articles relating to the evolution and improvement of PINNs in computational sciences and engineering, among others.

Content analysis was performed in the second stage to classify and analyze recent research across different disciplines and highlight potential challenges and limitations which could represent opportunities for future research.

### 3.3. Quantitative Synthesis (Meta-Analysis)

In a systematic review, quantitative synthesis is used to present statistical data. Typically, this is referred to as a meta-analysis. Table 1 below is the quantitative study character-

istics table, which consists of three groups. Journals are grouped by specialization, type, and the PINN method used in those papers.

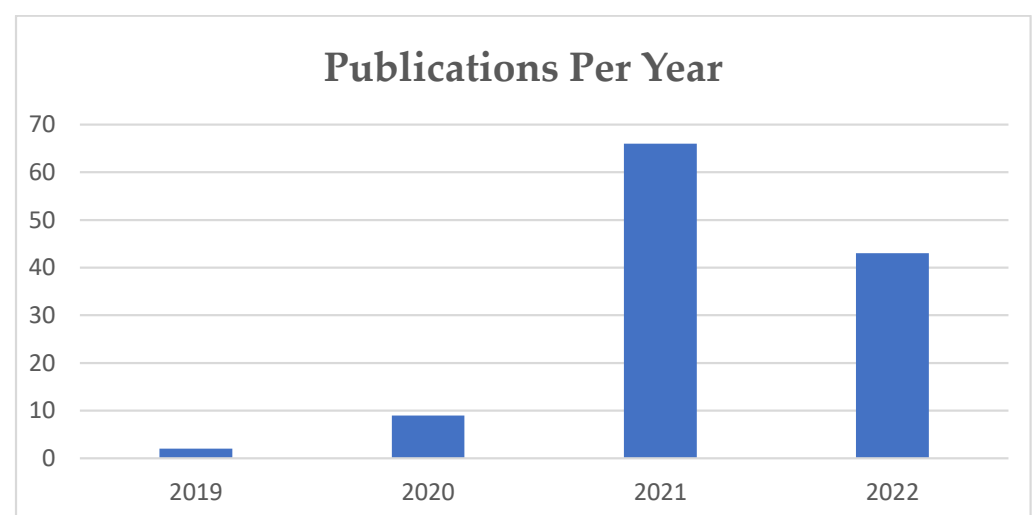
**Table 1.** Quantitative study characteristics.

Group	Population	Percentage (%)
Journal by Specialization		
1. Computer Science	29	24.167
2. Engineering	31	25.833
3. Mathematic	35	29.166
4. Physics	25	20.833
Journal by Type		
1. Conference Article	21	17.500
2. Journal Article	99	82.500
Journal by Methods		
1. Conventional PINNs	97	80.833
2. Extended PINNs	12	10.000
3. Hybrid PINNs	7	5.833
4. Minimized Loss PINNs	4	3.333

#### 4. Result of Bibliometric Analyses

The main objectives of this research, as mentioned earlier, were to assess the state of the art in this area of study using various bibliometric analyses and to classify the full range of eligibility requirements analyzed in the literature through metadata synthesis to produce a collection of general information on the extent to which the study of PINNs has changed over time.

Figure 4 illustrates physics-informed neural network evolution over the past three and a half years in terms of the number of publication relating to this area of study. Between 2019 and mid-2022, research in this field had been developing steadily, with a current peak in 2021. The trend shows that many researchers are busy finding solutions with PINNs, and some are also trying to optimize performance.



**Figure 4.** Publications per year.

Although PINNs represent a new field of research, they have attracted the interest of many researchers around the world. Figure 5 shows the countries with the most PINN publications. The United States, China, and European countries have the most publications.

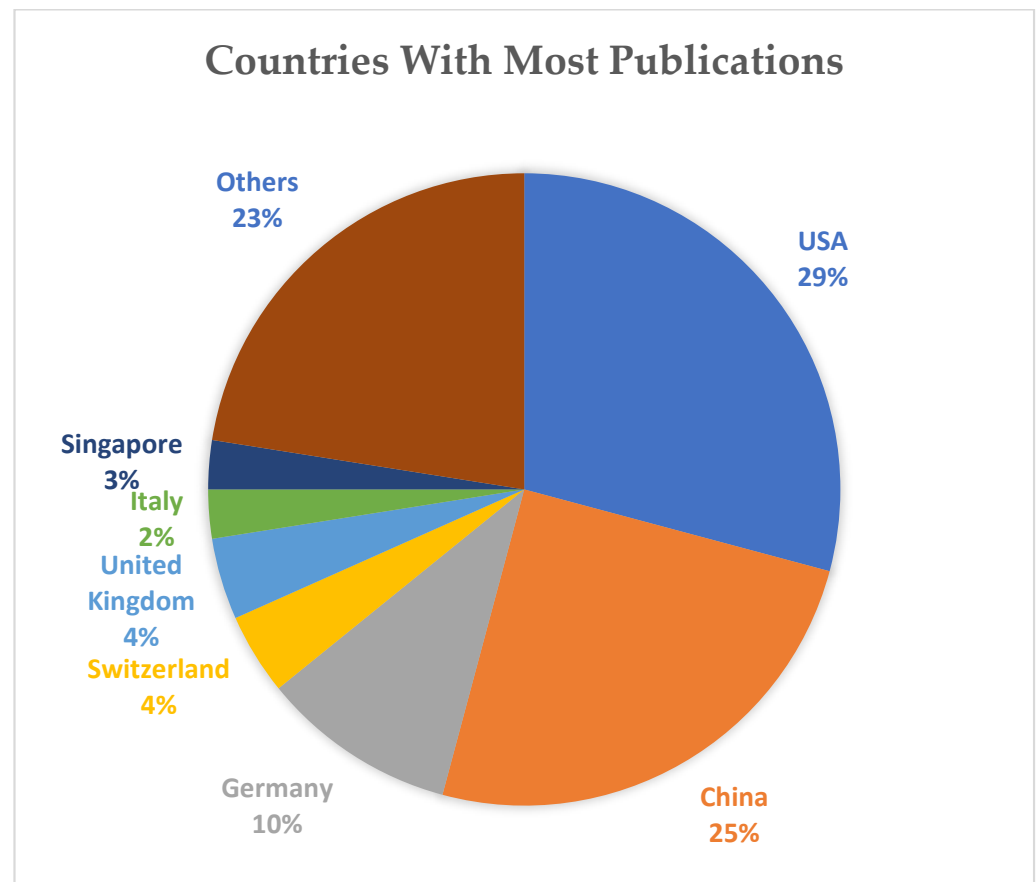
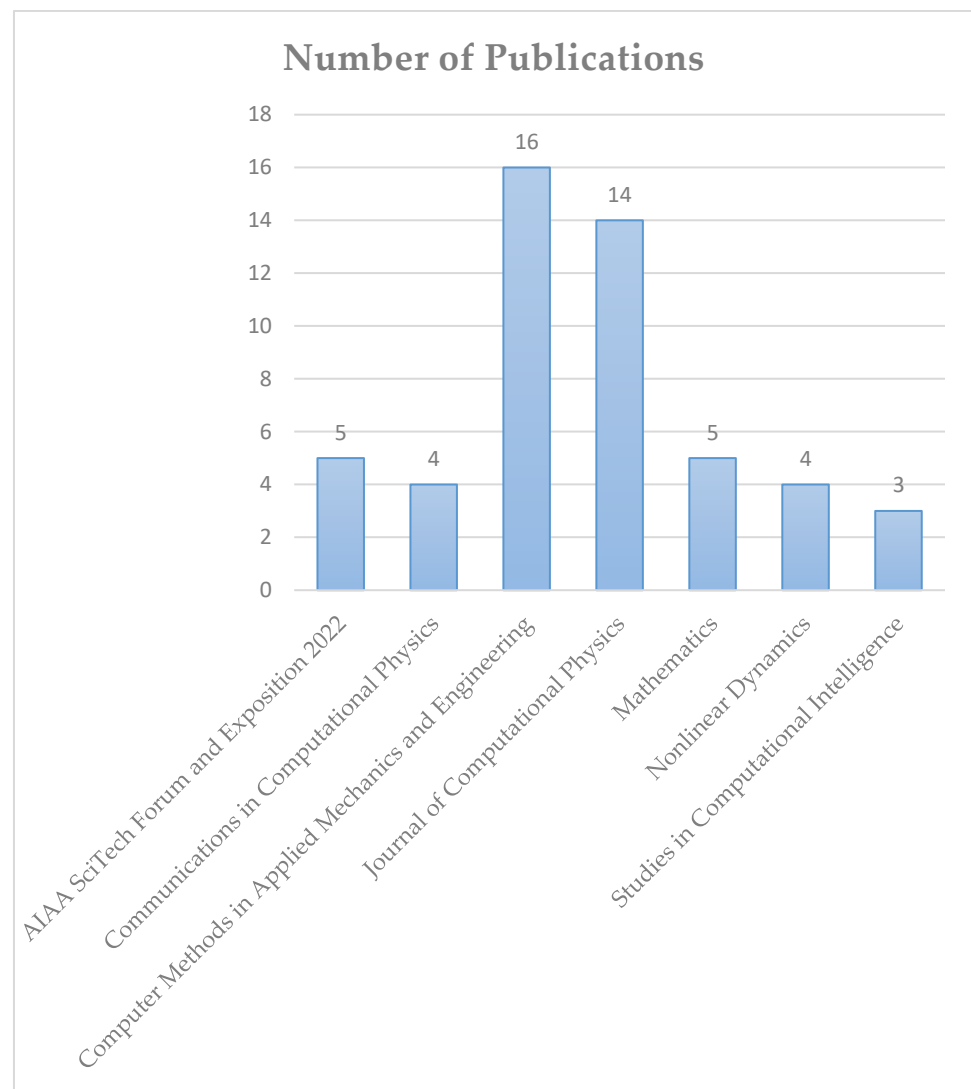


Figure 5. Countries with the most publications.

The seven most common publication sources represented in the final paper set are summarized in Figure 6. The data show that *Computer Methods in Applied Mechanics and Engineering* is the journal with the most articles in this area of study, followed by the *Journal of Computational Physics*. Many journals and conference proceedings have published numerous papers on PINNs. Out of 120 papers selected for this study, we considered journals with at least 3 published works in the list of article sources with the most publications.

The citation report of the studies from 2019 to mid-2022 is presented in Table 1. The most-cited authors are Raissi M., Perdikaris P., and Karniadakis G.E. [1], with 3442 citations as at the time of writing this paper. The most cited article's title is "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations". It was published by the *Journal of Computational Physics* and is indexed in the Scopus and Web of Science databases. These authors are considered pioneers of PINN research. After that, the second most-cited authors are Yang, Liu et al. [102] and Goswami et al. [103]. Their works were cited 183 and 177 times and published by the *Journal of Computational Physics* and *Theoretical and Applied Fracture Mechanics*, respectively. Further commonly cited studies are shown in Table 2 below.

Figure 7 illustrates journal publication sources with the most citations relating to physics-informed neural networks. No other journal publisher matched the number of citations that the *Journal of Computational Physics* received. The second most cited journals were *Communication in Computational Physics* and *Computer Methods in Applied Mechanics and Engineering*.



**Figure 6.** Journals with the most publications.

**Table 2.** Authors with the most citations.

Authors	Source Title	Number of Citations
Raissi et al. [1]	<i>Journal of Computational Physics</i>	3442
Costabal et al. [104]	<i>Frontiers in Physics</i>	122
Jagtap A.D. et al. [105]	<i>Communications in Computational Physics</i>	118
Meng, Xuhui et al. [106]	<i>Computer Methods in Applied Mechanics and Engineering</i>	143
Yang, Liu et al. [102]	<i>Journal of Computational Physics</i>	183
Haghighat E. et al. [107]	<i>Computer Methods in Applied Mechanics and Engineering</i>	161
Kharazmi E. et al. [108]	<i>Computer Methods in Applied Mechanics and Engineering</i>	111
Fang, Yin et al. [109]	<i>Nonlinear Dynamics</i>	29
Dourado A. et al. [110]	<i>Journal of Computing and Information Science in Engineering</i>	37
Shin Y. et al. [111]	<i>Communications in Computational Physics</i>	137

Table 2. Cont.

Authors	Source Title	Number of Citations
Zobeiry N. et al. [112]	<i>Engineering Applications of Artificial Intelligence</i>	52
Goswami et al. [103]	<i>Theoretical And Applied Fracture Mechanics</i>	177
Mehta, Pavan et al. [113]	<i>Fractional Calculus and Applied Analysis</i>	25
Colby et al. [18]	<i>Communications in Computational Physics</i>	54
Liu, Minliang et al. [114]	<i>Computer Methods in Applied Mechanics and Engineering</i>	26
Doan N.A.K. et al. [82]	<i>Journal of Computational Science</i>	28
Rao, Chengping et al. [92]	<i>Journal of Engineering Mechanics</i>	57
Pu, Juncai et al. [115]	<i>Nonlinear Dynamics</i>	21
Meng, Xuhui et al. [116]	<i>Journal of Computational Physics</i>	31
Li W. et al. [66]	<i>Computer Methods in Applied Mechanics and Engineering</i>	21

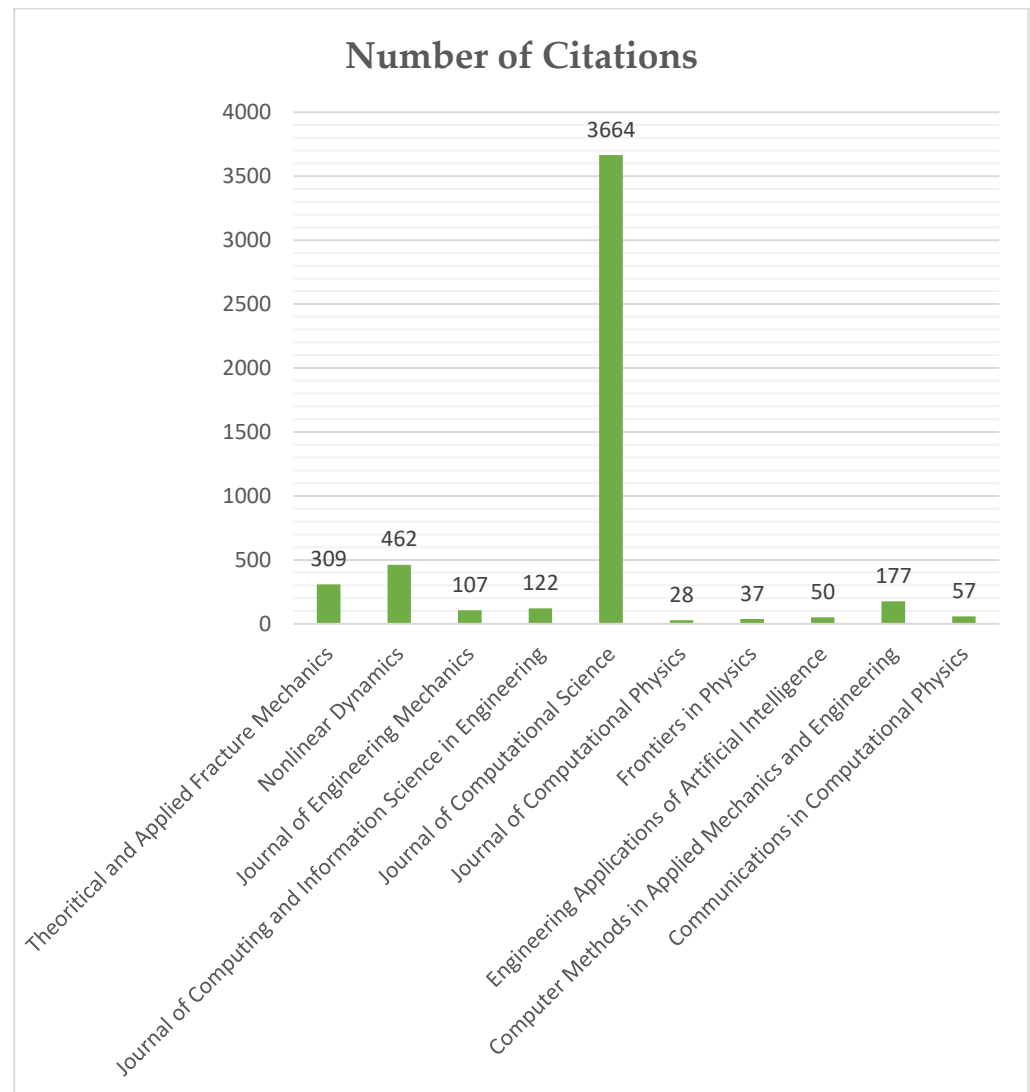


Figure 7. Journals with the most citations.



PINN research has garnered support from researchers and technology practitioners in recent years, as shown in Figure 4. Despite the popularity of PINNs, contributions to the field come from few countries around the globe. A review of the number of countries with the most publications related to PINNs using bibliometric analysis could spark the interest of scientists and technology practitioners from other countries and encourage them to collaborate and contribute, as many developed and developing countries are now competing for higher rankings in the lists of the best countries for scientific and technical research.

Consequently, evaluating journals with the most publications and authors with the most citations through bibliometric analysis can lead to more contributors to the literature because many journal publishers may emphasize PINN research. The number of citations an author receives from a particular article portrays how other people acknowledge their contribution and the impact of their research in the academic domain.

In this study, we found that highly cited journals have a unique novelty in terms of the area of discussion. They try to solve different scientific and technical problems with the techniques they propose. The publication from Raissi et al. [1] is the most influential article among all others. As we mentioned earlier, they are the pioneers of PINNs and the results of their work laid the foundation for PINN research.

In short, bibliometric analysis with a systematic literature search would give enthusiasts an insight into how far PINN research has progressed and where they should start their research.

#### 4.1. Newly Proposed PINN Methods

One of the main contributions of our study was finding a solution to the limitations of PINNs as mentioned by some authors. Our study also highlights several issues for future research. To focus on the research question, we evaluated the work of numerous authors who have worked on improving the performance of PINNs and found solutions to many of the limitations previously mentioned by different authors.

Although PINN architecture is built based on feedforward neural networks, due to the shortcomings of PINNs many authors have tried to extend PINNs by using different approaches, such as conservative PINNs (cPINNs), nonlocal PINNs (nPINNs), etc. In contrast, others have tried to combine it with other neural network techniques, expecting better performance and more precise results. This includes CNNs, RNNs, etc. Consequently, other authors have tried to boost performance by reducing loss to a minimal level. We tried to group newly proposed techniques into three categories: Extended PINNs, Hybrid PINNs, and Minimized Loss techniques.

##### 4.1.1. Extended PINNs

The studies under this category tried to extend the PINN using various techniques, such as domain and subdomain decomposition, to improve the generalization of PINNs. Other techniques, such as PINNs with ensemble methods, tried to expand the solution interval of PINNs to converge to the correct solution. Bayesian physics-informed neural networks (B-PINNs) are a type of Extended PINN that is more accurate and much faster than a simple PINN. The objectives and limitations of many more techniques are briefly discussed in Table 3 below.

**Table 3.** Proposed Extended PINNs.

Author	Objective(s)	Technique	Limitation(s)
Jagtap A.D. et al. [35]	The main goal of this study was to develop a unique conservative physics-informed neural network (cPINN) for solving complicated problems.	Conservative physics-informed neural network (cPINN)	Despite the parallelization of the cPINN, it cannot be used for parallel computation.
Jagtap A.D. et al. [117]	The main objective of this study was to introduce an XPINN model that improved the generalization capabilities of PINNs.	Extended physics-informed neural networks (XPINNs)	XPINNs enhance generalization in exceptional conditions. Decomposition results in less training data, which makes the model more likely to overfit and lose generalizability.
De Ryck et al. [14]	The main goal of this study was to precisely constrain the errors arising from the use of XPINNs to approximate incompressible Navier–Stokes equations.	PINN error estimates	The authors’ estimates in their experiment gave no indication of training errors.
G. Pang et al. [118]	This study aimed to extend PINNs to the inference of parameters and functions for integral equations, such as nonlocal Poisson and nonlocal turbulence models (nPINNs). A wide range of datasets must be adaptable to fit the nPINNs.	Nonlocal physics-informed neural networks (nPINNs)	nPINNs require more residual points. Increasing the number of discretization points, on the other hand, makes optimization more challenging and ineffective, and causes error stagnation.
Liu Yang et al. [102]	The aim of this study was to introduce a novel method that was designed for solving both forward and inverse nonlinear problems outlined by PDEs with noisy data, which aimed to be more accurate and much faster than a simple PINN.	Bayesian physics-informed neural networks (B-PINNs)	The proposed B-PINNs in this work were only tested in scenarios where data size was up to several hundreds, and no tests were performed with large datasets.
Ehsan Kharazmi et al. [108]	The purpose of this research was to bring together current developments in deep learning techniques for PDEs based on residuals of least-squares equations using a newly developed method.	Variational physics-informed neural networks (hp-VPINNs)	Although VPINN performance on inverse problems is encouraging, no comparison was made to classical approaches.
Juncai Pu et al. [115]	The goal of the study was to provide an improved PINN approach for localized wave solutions of the derivative nonlinear Schrödinger equation in complex space with faster convergence and optimum simulation performance.	Improved PINN method	Complex integrable equations were not really considered in this study.
Enrico Schiassi et al. [21]	The main objective of this study was to propose a novel model for providing solutions to problems with parametric differential equations (DEs) that is more accurate and robust.	Physics-informed neural network theory of functional connections (PINN-TFC)	The proposed technique cannot be applied to data-driven discovery of problems when solving ODEs using both a deterministic and probabilistic approach.
Rafiq et al. [119]	The main goal of this experiment was to propose a unique deep Fourier neural network that expands information using spectral feature combination and a Fourier neural operator as the principal component. The major objective of this experiment was to design a robust model architecture for reconstructing periodic flows with a small number of imperfect sensors by extending PINNs with forced truncated Fourier decomposition.	Deep spectral feature aggregation physics-informed neural network (DSFA-PINN)	Other mathematical functions, such as the Laplace transform coupled with a Fourier transform, as well as the conventional CNN, cannot be used to generalize models using this method.
Gaétan et al. [120]	The primary objective of this study was to present an Extended PINN method which is more effective and accurate in solving larger PDE problems.	Modal physics-informed neural networks (ModalPINNs)	The application of ModalPINNs is restricted to fluid mechanics only.
Colby et al. [18]		Adaptive physics informed neural networks	This study focused primarily on the problem of solving differential equations.
Katsiaryna et al. [121]	The objective of this experiment was to determine an acceptable time window for expanding the solution interval using an ensemble of PINNs.	PINNs with ensemble models	The ensemble algorithm seems to be more computationally intensive than the standard PINN and is not applicable to complex systems.

#### 4.1.2. Hybrid PINNs

Although PINNs are feedforward in nature, some researchers have tried to combine them with a variety of neural network architectures to overcome their limitations and improve overall performance. Parareal physics-informed neural networks (PPINNs) were proposed to improve high-level computational efficacy by using a small dataset. Hybrid physics-informed neural network (Hybrid PINNs) use convolutional neural networks to solve PDE problems. Physics-informed recurrent neural networks model an industrial process called a grey box. The objectives and limitations of Hybrid PINNs are highlighted in Table 4 below.

**Table 4.** Proposed Hybrid PINNs.

Author	Objective(s)	Technique	Limitation(s)
Meng et al. [106]	The main goal of this research was to introduce a new a hybrid technique that can exploit the high-level computational efficacy of training a neural network with small datasets to significantly speed up the time taken to find solutions to challenging physical problems.	Parareal physics-informed neural network (PPINN)	Domain decomposition of fundamental problems with huge spatial databases cannot be solved with PPINNs.
Zhiwei Fang et al. [33]	This paper aimed to present a Hybrid PINN for PDEs and a differential operator approximation for solving the PDEs using a convolutional neural network (CNN).	Hybrid physics-informed neural network (Hybrid PINN)	This Hybrid PINN is not applicable to nonlinear operators.
Lahariya M. [122]	The goal of this research was to propose a physics-informed neural network based on grey-box modeling methods for identifying energy buffers using a recurrent neural network.	Physics-informed recurrent neural networks	The proposed model was not validated with real-world industrial processes.
Wenqian Chen et al. [11]	The main goal of this research was to develop a reduced-order model that uses high-accuracy snapshots to generate reduced basis information from the accurate network while reducing the weighted sum of residual losses from the reduced-order equation.	Physics-reinforced neural network (PRNN)	The reduced basis set must be small to outperform the Proper Orthogonal Decomposition–Galerkin (POD–G) method in terms of accuracy, as the numerical results of the experiment showed.
Xiaoping Zhang [123]	The main objective of this study was to develop a novel method for solving groundwater flow equations using deep learning techniques.	Ground Water-PINN (GW-PINN)	The proposed model cannot be used to predict groundwater flow in more complex and larger areas.
Dourado et al. [110]	The major goal of this experiment was to develop a hybrid technique for missing physics estimates in cumulative damage models by combining data-driven and physics-informed layers in deep neural networks.	PINNs for missing physics	Even if the proposed additional levels are used to initialize the neural network, suboptimal setting of these parameters may lead to the failure of the training.
Mingyuan Yang [124]	The goal of this experiment was to develop a new hybrid model for uncertain forward and inverse PDE problems.	Multi-Output physics-informed neural network (MO-PINN)	The proposed method cannot be used to solve problems involving multi-fidelity data.

#### 4.1.3. Minimized Loss Techniques

The newly proposed methods under this category attempt to minimize loss using different techniques. For instance, the new reptile initialization-based physics-informed neural network (NRPINN) uses many sample tasks from parameterized PDEs. It modifies the loss penalty term so the model can adopt the initialization parameters of related tasks using supervised, unsupervised, and semi-supervised learning. Consequently, the physics-informed and physics-penalized neural network model (PI-PP-NN) expresses many physical constraints and integrates the regulating physical laws into its loss function. The objectives and limitations of newly proposed techniques are briefly discussed in Table 5 below.

**Table 5.** Proposed PINN techniques using Minimized Loss.

Author	Objective(s)	Technique	Limitation(s)
Apostolos F. et al. [125]	The main goal of this study was to provide a gradient-based meta-learning method for offline discovery that uses data from task distributions created using parameterized PDEs with numerous benchmarks to meta-learn PINN loss functions.	Meta-learning PINN loss functions	Optimizing the performance of methods such as RMSProp and Adam for handling inner optimizers with memory was not considered in this experiment.
Liu X. et al. [8]	The major objective of this experiment was to use multiple sample tasks from parameterized PDEs and modify the loss penalty term to introduce a novel method that depends on labeled data.	New reptile initialization-based physics-informed neural network (NRPINN)	NRPINNs cannot be used to solve problems in the absence of prior knowledge.
Habib et al. [126]	The main goal of this experiment was to develop a model that expresses physical constraints and integrates the regulating physical laws into its loss function (physics-informed), which the model penalizes when they are violated (physics-penalized).	Physics-informed and physics-penalized neural network model (PI-PP-NN)	The proposed model can only be used to create friction pendulum bearings. For any other isolation system, the theoretical basis must be adapted accordingly before it can be used for design.
Zixue Xiang [127]	The main goal of this experiment was to develop a technique that allows PINNs to perfectly and efficiently learn PDEs using Gaussian probabilistic models.	Loss-balanced physics-informed neural networks (lbPINNs)	In this experiment, the adaptive weight of PDE loss gradually decreased. Therefore, a theoretical investigation of this paradigm is necessary to increase the robustness and scalability of the technique.

## 5. Future Research Direction

We previously pointed out various limitations of PINNs that were highlighted by different authors and later discussed the newly proposed techniques used to solve most of the mentioned problems. We have also outlined the limitations of the newly proposed PINN methods. The focus of our future research will be more on the limitations of the newly proposed techniques. Since they have addressed most of the PINN limitations, the limitations of some selected articles are discussed because of their significance. One of the main benefits of using the cPINNs introduced by [35] to solve complicated problems is their capacity for parallelization, which effectively lowers training costs; however, they cannot be used for parallel computation. In future research, cPINNs may be extended for use in parallel computation.

The enhanced PINN technique proposed by [115] improves neural network performance. Through numerous trainable parameters in the activation function, they noted that an important issue for the future was considering how to integrate machine learning using integrable systems theory more fully and construct substantial integrable deep learning algorithms.

A new approach presented by [21] can also be applied to data-driven discoveries and solutions of parametric differential equations. Future research will focus on extending this

method to data-driven problem discovery for solving ODEs using both deterministic and probabilistic methods.

Noting the nature of optimization when training PINNs, the capacity to perform uncertainty quantification (UQ) on physical systems was highlighted by [1]. The PINN framework implemented by [92] could be extended to provide these features.

Despite the numerous desirable features of the hybrid approach proposed by [110], their findings proved that the methodology has a shortcoming that could be considered as a topic of future research. Their research has limitations regarding training results, as they were proven to be very responsive to the initial hyperparameters of the neural network; even if the proposed auxiliary planes are used to initialize the neural network, the poor initialization of these parameters can potentially hinder training. Therefore, optimization with a wide range of initial hyperparameter values is still required.

The recent work of Liu X. et al. [8] has one limitation. As a new reptilian initialization learning task, NRPINN needs prior information, especially higher- and zero-order information. As a result, NRPINN is not suitable for handling problems where previous information is not available. Using transfer learning from related efforts to acquire an initialization may be another technique for improving the performance of PINNs in future studies.

A new hybrid technique proposed by [106] cannot be used to solve the domain decomposition of basic problems with big spatial databases. Extending this work to support large spatial databases could be a possible area of future research. The Ground Water-PINN (GW-PINN) technique proposed by [123] can be extended to predict groundwater flow in more complex and larger areas.

The work of [124] could be extended to problems with multi-fidelity data in future research. Additionally, their approach could also be stretched to time-dependent problems because time dimension is comparable to space dimensions from an implementation standpoint; since early uncertainties are transmitted and magnified over time, distinct behavior in relation to the uncertainty correlation may emerge.

## 6. Conclusions

The review and bibliometric analysis of the published literature revealed several limitations of PINNs. We discovered that a significant amount of experimental research in the 120 peer-reviewed articles used conventional PINNs to solve various scientific and engineering problems. In contrast, other studies developed new methods to overcome the limitations of PINNs and achieve higher performance results.

Data evaluation was an essential and significant step in the review. Based on the relative number of PINN-related journals, two reputable databases were chosen: Scopus and Web of Science. The first PINN papers were published in early 2019. For this reason, the papers used in this study are from 2019 to mid-2022.

We discussed the objectives, methodologies, and limitations of the newly proposed PINN techniques. Despite the feedforward nature of PINNs, we have found several articles that combined them with other neural network architectures. While some extended the conventional PINNs, others tried to improve performance by using different techniques for reducing loss. We have also identified several potential research directions for PINNs based on the limitations of proposed solutions to increasing prediction power and optimizing performance.

As part of our contribution to the literature, we intend to implement a new model that combines PINNs with either a graph neural network or a recurrent neural network using a time series dataset.

**Supplementary Materials:** The following supporting information can be downloaded at: <https://www.mdpi.com/article/10.3390/bdcc6040140/s1>, the PRISMA 2020 Checklist [128].



**Author Contributions:** Writing—original draft preparation, A.C.I. and Z.K.L.; writing—review and editing, Z.K.L., A.C.I. and D.T.C.L.; supervision, A.C.I., H.Y. and D.T.C.L.; funding acquisition, H.Y. and A.C.I. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work is funded by Universiti Brunei Darussalam, Brunei under Grant ref: UBD/RSCH/1.11/FICBF(b)/2020/004.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** The data were extracted from the Scopus and Web of Science websites.

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

## References

1. Raissi, M.; Perdikaris, P.; Karniadakis, G.E. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *J. Comput. Phys.* **2019**, *378*, 686–707. [\[CrossRef\]](#)
2. Hu, Z.; Jagtap, A.D.; Karniadakis, G.E.; Kawaguchi, K. When Do Extended Physics-Informed Neural Networks (XPINNs) Improve Generalization? *arXiv* **2021**, arXiv:2109.09444. [\[CrossRef\]](#)
3. Shukla, K.; Jagtap, A.D.; Karniadakis, G.E. Parallel physics-informed neural networks via domain decomposition. *J. Comput. Phys.* **2021**, *447*, 110683. [\[CrossRef\]](#)
4. Ang, E.; Ng, B.F. Physics-Informed Neural Networks for Flow Around Airfoil. In *AIAA SCITECH 2022 Forum*; American Institute of Aeronautics and Astronautics: Fairfax, VA, USA, 2021. [\[CrossRef\]](#)
5. Gnanasambandam, R.; Shen, B.; Chung, J.; Yue, X. Self-scalable Tanh (Stan): Faster Convergence and Better Generalization in Physics-informed Neural Networks. *arXiv* **2022**, arXiv:2204.12589.
6. Cai, S.; Wang, Z.; Wang, S.; Perdikaris, P.; Karniadakis, G.E. Physics-Informed Neural Networks for Heat Transfer Problems. *J. Heat Transf.* **2021**, *143*, 060801. [\[CrossRef\]](#)
7. Chiu, P.-H.; Wong, J.C.; Ooi, C.; Dao, M.H.; Ong, Y.-S. CAN-PINN: A fast physics-informed neural network based on coupled-automatic-numerical differentiation method. *Comput. Methods Appl. Mech. Eng.* **2022**, *395*, 114909. [\[CrossRef\]](#)
8. Liu, X.; Zhang, X.; Peng, W.; Zhou, W.; Yao, W. A novel meta-learning initialization method for physics-informed neural networks. *arXiv* **2022**, arXiv:2107.10991. [\[CrossRef\]](#)
9. Yang, S.; Chen, H.-C.; Wu, C.-H.; Wu, M.-N.; Yang, C.-H. Forecasting of the Prevalence of Dementia Using the LSTM Neural Network in Taiwan. *Mathematics* **2021**, *9*, 488. [\[CrossRef\]](#)
10. Huang, B.; Wang, J. Applications of Physics-Informed Neural Networks in Power Systems—A Review. *IEEE Trans. Power Syst.* **2022**, *1*. [\[CrossRef\]](#)
11. Chen, W.; Wang, Q.; Hesthaven, J.S.; Zhang, C. Physics-informed machine learning for reduced-order modeling of nonlinear problems. *J. Comput. Phys.* **2021**, *446*, 110666. [\[CrossRef\]](#)
12. Chen, Z.; Liu, Y.; Sun, H. Physics-informed learning of governing equations from scarce data. *Nat. Commun.* **2021**, *12*, 6136. [\[CrossRef\]](#)
13. Karakusak, M.Z.; Kivrak, H.; Ates, H.F.; Ozdemir, M.K. RSS-Based Wireless LAN Indoor Localization and Tracking Using Deep Architectures. *Big Data Cogn. Comput.* **2022**, *6*, 84. [\[CrossRef\]](#)
14. De Ryck, T.; Jagtap, A.D.; Mishra, S. Error estimates for physics informed neural networks approximating the Navier-Stokes equations. *arXiv* **2022**, arXiv:2203.09346.
15. Zhai, H.; Sands, T. Controlling Chaos in Van Der Pol Dynamics Using Signal-Encoded Deep Learning. *Mathematics* **2022**, *10*, 453. [\[CrossRef\]](#)
16. Zhang, T.; Xu, H.; Guo, L.; Feng, X. A non-intrusive neural network model order reduction algorithm for parameterized parabolic PDEs. *Comput. Math. Appl.* **2022**, *119*, 59–67. [\[CrossRef\]](#)
17. Ankita; Rani, S.; Singh, A.; Elkamchouchi, D.H.; Noya, I.D. Lightweight Hybrid Deep Learning Architecture and Model for Security in IIOT. *Appl. Sci.* **2022**, *12*, 6442. [\[CrossRef\]](#)
18. Wight, C.L.; Zhao, J. Solving Allen-Cahn and Cahn-Hilliard Equations using the Adaptive Physics Informed Neural Networks. *arXiv* **2020**, arXiv:2007.04542.
19. Rasht-Behesht, M.; Huber, C.; Shukla, K.; Karniadakis, G.E. Physics-Informed Neural Networks (PINNs) for Wave Propagation and Full Waveform Inversions. *J. Geophys. Res. Solid Earth* **2022**, *127*, e2021JB023120. [\[CrossRef\]](#)
20. Nasiri, P.; Dargazany, R. Reduced-PINN: An Integration-Based Physics-Informed Neural Networks for Stiff ODEs. *arXiv* **2020**, arXiv:2208.12045.
21. Schiassi, E.; De Florio, M.; D'Ambrosio, A.; Mortari, D.; Furfaro, R. Physics-Informed Neural Networks and Functional Interpolation for Data-Driven Parameters Discovery of Epidemiological Compartmental Models. *Mathematics* **2021**, *9*, 2069. [\[CrossRef\]](#)
22. Zhang, Z.; Li, Y.; Zhou, W.; Chen, X.; Yao, W.; Zhao, Y. TONR: An exploration for a novel way combining neural network with topology optimization. *Comput. Methods Appl. Mech. Eng.* **2021**, *386*, 114083. [\[CrossRef\]](#)

23. Wang, S.; Teng, Y.; Perdikaris, P. Understanding and mitigating gradient pathologies in physics-informed neural networks. *arXiv* **2020**, arXiv:2001.04536. [[CrossRef](#)]
24. Fujita, K. Physics-Informed Neural Network Method for Space Charge Effect in Particle Accelerators. *IEEE Access* **2021**, *9*, 164017–164025. [[CrossRef](#)]
25. Yu, J.; de Antonio, A.; Villalba-Mora, E. Deep Learning (CNN, RNN) Applications for Smart Homes: A Systematic Review. *Computers* **2022**, *11*, 26. [[CrossRef](#)]
26. Dwivedi, V.; Srinivasan, B. A Normal Equation-Based Extreme Learning Machine for Solving Linear Partial Differential Equations. *J. Comput. Inf. Sci. Eng.* **2021**, *22*, 014502. [[CrossRef](#)]
27. Haghighat, E.; Amini, D.; Juanes, R. Physics-informed neural network simulation of multiphase poroelasticity using stress-split sequential training. *Comput. Methods Appl. Mech. Eng.* **2021**, *397*, 115141. [[CrossRef](#)]
28. Berg, J.; Nyström, K. A unified deep artificial neural network approach to partial differential equations in complex geometries. *Neurocomputing* **2018**, *317*, 28–41. [[CrossRef](#)]
29. Mahesh, R.B.; Leandro, J.; Lin, Q. Physics informed neural network for spatial-temporal flood forecasting. In *Climate Change and Water Security*; Lecture Notes in Civil Engineering; Springer Nature Singapore Pte Ltd.: Singapore, 2022; Volume 178. [[CrossRef](#)]
30. Ngo, P.; Tejedor, M.; Tayefi, M.; Chomutare, T.; Godtliebsen, F. Risk-Averse Food Recommendation Using Bayesian Feedforward Neural Networks for Patients with Type 1 Diabetes Doing Physical Activities. *Appl. Sci.* **2020**, *10*, 8037. [[CrossRef](#)]
31. Henkes, A.; Wessels, H.; Mahnken, R. Physics informed neural networks for continuum micromechanics. *Comput. Methods Appl. Mech. Eng.* **2022**, *393*, 114790. [[CrossRef](#)]
32. Patel, R.G.; Manickam, I.; Trask, N.A.; Wood, M.A.; Lee, M.; Tomas, I.; Cyr, E.C. Thermodynamically consistent physics-informed neural networks for hyperbolic systems. *J. Comput. Phys.* **2020**, *449*, 110754. [[CrossRef](#)]
33. Fang, Z. A High-Efficient Hybrid Physics-Informed Neural Networks Based on Convolutional Neural Network. *IEEE Trans. Neural Networks Learn. Syst.* **2021**, *33*, 5514–5526. [[CrossRef](#)]
34. Lawal, Z.K.; Yassin, H.; Zakari, R.Y. Flood Prediction Using Machine Learning Models: A Case Study of Kebbi State Nigeria. In *Proceedings of the 2021 IEEE Asia-Pacific Conference on Computer Science and Data Engineering (CSDE)*, Brisbane, Australia, 8–10 December 2021. [[CrossRef](#)]
35. Jagtap, A.D.; Kharazmi, E.; Karniadakis, G.E. Conservative physics-informed neural networks on discrete domains for conservation laws: Applications to forward and inverse problems. *Comput. Methods Appl. Mech. Eng.* **2020**, *365*, 113028. [[CrossRef](#)]
36. Mao, Z.; Jagtap, A.D.; Karniadakis, G.E. Physics-informed neural networks for high-speed flows. *Comput. Methods Appl. Mech. Eng.* **2020**, *360*, 112789. [[CrossRef](#)]
37. Bihlo, A.; Popovych, R.O. Physics-informed neural networks for the shallow-water equations on the sphere. *J. Comput. Phys.* **2022**, *456*, 111024. [[CrossRef](#)]
38. Lagaris, I.; Likas, A.; Fotiadis, D. Artificial neural networks for solving ordinary and partial differential equations. *IEEE Trans. Neural Networks* **1998**, *9*, 987–1000. [[CrossRef](#)]
39. Zjavka, L. Construction and adjustment of differential polynomial neural network. *J. Eng. Comput. Innov.* **2011**, *2*, 40–50.
40. Zjavka, L. Approximation of multi-parametric functions using the differential polynomial neural network. *Math. Sci.* **2013**, *7*, 33. [[CrossRef](#)]
41. Zjavka, L.; Snasel, V. Composing and Solving General Differential Equations Using Extended Polynomial Networks. In *Proceedings of the 2015 International Conference on Intelligent Networking and Collaborative Systems, IEEE INCOS 2015*, Taipei, Taiwan, 2–4 September 2015; pp. 110–115. [[CrossRef](#)]
42. Raissi, M.; Karniadakis, G.E. Hidden physics models: Machine learning of nonlinear partial differential equations. *J. Comput. Phys.* **2018**, *357*, 125–141. [[CrossRef](#)]
43. Raissi, M.; Perdikaris, P.; Karniadakis, G.E. Physics Informed Deep Learning (Part I): Data-driven Solutions of Nonlinear Partial Differential Equations. *arXiv* **2017**, arXiv:1711.10561.
44. Raissi, M.; Perdikaris, P.; Karniadakis, G.E. Multistep Neural Networks for Data-driven Discovery of Nonlinear Dynamical Systems. *arXiv* **2018**, arXiv:1801.01236.
45. Raissi, M. Deep Hidden Physics Models: Deep Learning of Nonlinear Partial Differential Equations. 2018. Available online: <http://jmlr.org/papers/v19/18-046.html> (accessed on 3 June 2022).
46. Raissi, M.; Perdikaris, P.; Karniadakis, G.E. Numerical Gaussian Processes for Time-dependent and Non-linear Partial Differential Equations. *arXiv* **2017**, arXiv:1703.10230.
47. Karniadakis, G.E.; Kevrekidis, I.G.; Lu, L.; Perdikaris, P.; Wang, S.; Yang, L. Physics-informed machine learning. *Nat. Rev. Phys.* **2021**, *3*, 422–440. [[CrossRef](#)]
48. Lazovskaya, T.; Malykhina, G.; Tarkhov, D. Physics-Based Neural Network Methods for Solving Parameterized Singular Perturbation Problem. *Computation* **2021**, *9*, 97. [[CrossRef](#)]
49. Bati, G.F.; Singh, V.K. Nadal: A neighbor-aware deep learning approach for inferring interpersonal trust using smartphone data. *Computers* **2021**, *10*, 3. [[CrossRef](#)]
50. Klyuchinskiy, D.; Novikov, N.; Shishlenin, M. A Modification of Gradient Descent Method for Solving Coefficient Inverse Problem for Acoustics Equations. *Computation* **2020**, *8*, 73. [[CrossRef](#)]
51. Li, J.; Zheng, L. DEEPWAVE: Deep Learning based Real-time Water Wave Simulation. Available online: <https://jinningli.cn/cv/DeepWavePaper.pdf> (accessed on 25 May 2022).

52. Nascimento, R.G.; Fricke, K.; Viana, F.A. A tutorial on solving ordinary differential equations using Python and hybrid physics-informed neural network. *Eng. Appl. Artif. Intell.* **2020**, *96*, 103996. [\[CrossRef\]](#)
53. Cheng, Y.; Huang, Y.; Pang, B.; Zhang, W. ThermalNet: A deep reinforcement learning-based combustion optimization system for coal-fired boiler. *Eng. Appl. Artif. Intell.* **2018**, *74*, 303–311. [\[CrossRef\]](#)
54. D'Ambrosio, A.; Schiassi, E.; Curti, F.; Furfaro, R. Pontryagin Neural Networks with Functional Interpolation for Optimal Intercept Problems. *Mathematics* **2021**, *9*, 996. [\[CrossRef\]](#)
55. Raissi, M.; Perdikaris, P.; Karniadakis, G.E. Inferring solutions of differential equations using noisy multi-fidelity data. *J. Comput. Phys.* **2017**, *335*, 736–746. [\[CrossRef\]](#)
56. Lawal, Z.K.; Yassin, H.; Zakari, R.Y. Stock Market Prediction using Supervised Machine Learning Techniques: An Overview. In Proceedings of the 2020 IEEE Asia-Pacific Conference on Computer Science and Data Engineering (CSDE), Gold Coast, Australia, 16–18 December 2020; pp. 1–6. [\[CrossRef\]](#)
57. Deng, R.; Duzhin, F. Topological Data Analysis Helps to Improve Accuracy of Deep Learning Models for Fake News Detection Trained on Very Small Training Sets. *Big Data Cogn. Comput.* **2022**, *6*, 74. [\[CrossRef\]](#)
58. Hornik, K.; Stinchcombe, M.; White, H. Multilayer feedforward networks are universal approximators. *Neural Netw.* **1989**, *2*, 359–366. [\[CrossRef\]](#)
59. Dong, S.; Li, Z. Local extreme learning machines and domain decomposition for solving linear and nonlinear partial differential equations. *Comput. Methods Appl. Mech. Eng.* **2021**, *387*, 114129. [\[CrossRef\]](#)
60. Alavizadeh, H.; Alavizadeh, H.; Jang-Jaccard, J. Deep Q-Learning Based Reinforcement Learning Approach for Network Intrusion Detection. *Computers* **2022**, *11*, 41. [\[CrossRef\]](#)
61. Arzani, A.; Dawson, S.T.M. Data-driven cardiovascular flow modelling: Examples and opportunities. *J. R. Soc. Interface* **2020**, *18*, 20200802. [\[CrossRef\]](#)
62. SBerrone, S.; Della Santa, F.; Mastropietro, A.; Pieraccini, S.; Vaccarino, F. Graph-Informed Neural Networks for Regressions on Graph-Structured Data. *Mathematics* **2022**, *10*, 786. [\[CrossRef\]](#)
63. Gutiérrez-Muñoz, M.; Coto-Jiménez, M. An Experimental Study on Speech Enhancement Based on a Combination of Wavelets and Deep Learning. *Computation* **2022**, *10*, 102. [\[CrossRef\]](#)
64. Mousavi, S.M.; Ghasemi, M.; Dehghan Manshadi, M.; Mosavi, A. Deep Learning for Wave Energy Converter Modeling Using Long Short-Term Memory. *Mathematics* **2021**, *9*, 871. [\[CrossRef\]](#)
65. Viana, F.A.; Nascimento, R.G.; Dourado, A.; Yucesan, Y.A. Estimating model inadequacy in ordinary differential equations with physics-informed neural networks. *Comput. Struct.* **2021**, *245*, 106458. [\[CrossRef\]](#)
66. Li, W.; Bazant, M.Z.; Zhu, J. A physics-guided neural network framework for elastic plates: Comparison of governing equations-based and energy-based approaches. *Comput. Methods Appl. Mech. Eng.* **2021**, *383*, 113933. [\[CrossRef\]](#)
67. Reyes, B.; Howard, A.A.; Perdikaris, P.; Tartakovsky, A.M. Learning unknown physics of non-Newtonian fluids. *Phys. Rev. Fluids* **2020**, *6*, 073301. [\[CrossRef\]](#)
68. Zhu, J.-A.; Jia, Y.; Lei, J.; Liu, Z. Deep Learning Approach to Mechanical Property Prediction of Single-Network Hydrogel. *Mathematics* **2021**, *9*, 2804. [\[CrossRef\]](#)
69. Rodrigues, P.J.; Gomes, W.; Pinto, M.A. DeepWings<sup>®</sup>: Automatic Wing Geometric Morphometrics Classification of Honey Bee (*Apis mellifera*) Subspecies Using Deep Learning for Detecting Landmarks. *Big Data Cogn. Comput.* **2022**, *6*, 70. [\[CrossRef\]](#)
70. Ji, W.; Qiu, W.; Shi, Z.; Pan, S.; Deng, S. Stiff-PINN: Physics-Informed Neural Network for Stiff Chemical Kinetics. *J. Phys. Chem. A* **2021**, *125*, 8098–8106. [\[CrossRef\]](#) [\[PubMed\]](#)
71. James, S.C.; Zhang, Y.; O'Donncha, F. A machine learning framework to forecast wave conditions. *Coast. Eng.* **2018**, *137*, 1–10. [\[CrossRef\]](#)
72. Hu, X.; Buris, N.E. A Deep Learning Framework for Solving Rectangular Waveguide Problems. In Proceedings of the Asia-Pacific Microwave Conference Proceedings, APMC, Hong Kong, 8–11 December 2020; pp. 409–411. [\[CrossRef\]](#)
73. Goodfellow, I.; Bengio, Y.; Courville, A. *Deep Learning*; The MIT Press: Cambridge, MA, USA, 2016.
74. Lim, S.; Shin, J. Application of a Deep Neural Network to Phase Retrieval in Inverse Medium Scattering Problems. *Computation* **2021**, *9*, 56. [\[CrossRef\]](#)
75. Wang, D.-L.; Sun, Q.-Y.; Li, Y.-Y.; Liu, X.-R. Optimal Energy Routing Design in Energy Internet with Multiple Energy Routing Centers Using Artificial Neural Network-Based Reinforcement Learning Method. *Appl. Sci.* **2019**, *9*, 520. [\[CrossRef\]](#)
76. Su, B.; Xu, C.; Li, J. A Deep Neural Network Approach to Solving for Seal's Type Partial Integro-Differential Equation. *Mathematics* **2022**, *10*, 1504. [\[CrossRef\]](#)
77. Seo, J.-K. A pretraining domain decomposition method using artificial neural networks to solve elliptic PDE boundary value problems. *Sci. Rep.* **2022**, *12*, 13939. [\[CrossRef\]](#)
78. Mishra, S.; Molinaro, R. Estimates on the generalization error of Physics Informed Neural Networks (PINNs) for approximating a class of inverse problems for PDEs. *arXiv* **2020**, arXiv:2007.01138.
79. Li, Y.; Wang, J.; Huang, Z.; Gao, R.X. Physics-informed meta learning for machining tool wear prediction. *J. Manuf. Syst.* **2022**, *62*, 17–27. [\[CrossRef\]](#)
80. Arzani, A.; Wang, J.-X.; D'Souza, R.M. Uncovering near-wall blood flow from sparse data with physics-informed neural networks. *Phys. Fluids* **2021**, *33*, 071905. [\[CrossRef\]](#)

81. Glorot, X.; Bengio, Y. Understanding the difficulty of training deep feedforward neural networks. In Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, Sardinia, Italy, 13–15 May 2010; Volume 9, pp. 249–256. Available online: <https://proceedings.mlr.press/v9/glorot10a.html> (accessed on 17 June 2022).
82. Doan, N.; Polifke, W.; Magri, L. Physics-informed echo state networks. *J. Comput. Sci.* **2020**, *47*, 101237. [\[CrossRef\]](#)
83. Falas, S.; Konstantinou, C.; Michael, M.K. Special Session: Physics-Informed Neural Networks for Securing Water Distribution Systems. In Proceedings of the IEEE International Conference on Computer Design: VLSI in Computers and Processors, Hartford, CT, USA, 18–21 October 2020; pp. 37–40. [\[CrossRef\]](#)
84. Filgöz, A.; Demirezen, G.; Demirezen, M.U. Applying Novel Adaptive Activation Function Theory for Launch Acceptability Region Estimation with Neural Networks in Constrained Hardware Environments: Performance Comparison. In Proceedings of the 2021 IEEE/AIAA 40th Digital Avionics Systems Conference (DASC), San Antonio, TX, USA, 3–7 October 2021; pp. 1–10. [\[CrossRef\]](#)
85. Fülöp, A.; Horváth, A. End-to-End Training of Deep Neural Networks in the Fourier Domain. *Mathematics* **2022**, *10*, 2132. [\[CrossRef\]](#)
86. Fang, Z.; Zhan, J. A Physics-Informed Neural Network Framework for PDEs on 3D Surfaces: Time Independent Problems. *IEEE Access* **2020**, *8*, 26328–26335. [\[CrossRef\]](#)
87. Markidis, S. The Old and the New: Can Physics-Informed Deep-Learning Replace Traditional Linear Solvers? *arXiv* **2021**, arXiv:2103.09655. [\[CrossRef\]](#)
88. Baydin, A.G.; Pearlmutter, B.A.; Radul, A.A.; Siskind, J.M. Automatic differentiation in machine learning: A survey. *arXiv* **2015**, arXiv:1502.05767.
89. Niaki, S.A.; Haghighat, E.; Campbell, T.; Poursartip, A.; Vaziri, R. Physics-informed neural network for modelling the thermo-chemical curing process of composite-tool systems during manufacture. *Comput. Methods Appl. Mech. Eng.* **2021**, *384*, 113959. [\[CrossRef\]](#)
90. Li, Y.; Xu, L.; Ying, S. DWNN: Deep Wavelet Neural Network for Solving Partial Differential Equations. *Mathematics* **2022**, *10*, 1976. [\[CrossRef\]](#)
91. De Wolff, T.; Carrillo, H.; Martí, L.; Sanchez-Pi, N. Assessing Physics Informed Neural Networks in Ocean Modelling and Climate Change Applications. In Proceedings of the AI: Modeling Oceans and Climate Change Workshop at ICLR 2021, Santiago, Chile, 7 May 2021; Available online: <https://hal.inria.fr/hal-03262684> (accessed on 17 June 2022).
92. Rao, C.; Sun, H.; Liu, Y. Physics informed deep learning for computational elastodynamics without labeled data. *arXiv* **2020**, arXiv:2006.08472. [\[CrossRef\]](#)
93. Liu, X.; Almekkawy, M. Ultrasound Computed Tomography using physical-informed Neural Network. In Proceedings of the 2021 IEEE International Ultrasonics Symposium (IUS), Xi'an, China, 11–16 September 2021; pp. 1–4. [\[CrossRef\]](#)
94. Vitanov, N.K.; Dimitrova, Z.I.; Vitanov, K.N. On the Use of Composite Functions in the Simple Equations Method to Obtain Exact Solutions of Nonlinear Differential Equations. *Computation* **2021**, *9*, 104. [\[CrossRef\]](#)
95. Guo, Y.; Cao, X.; Liu, B.; Gao, M. Solving Partial Differential Equations Using Deep Learning and Physical Constraints. *Appl. Sci.* **2020**, *10*, 5917. [\[CrossRef\]](#)
96. Li, J.; Tartakovsky, A.M. Physics-informed Karhunen-Loève and neural network approximations for solving inverse differential equation problems. *J. Comput. Phys.* **2022**, *462*, 111230. [\[CrossRef\]](#)
97. Qureshi, M.; Khan, N.; Qayyum, S.; Malik, S.; Sanil, H.; Ramayah, T. Classifications of Sustainable Manufacturing Practices in ASEAN Region: A Systematic Review and Bibliometric Analysis of the Past Decade of Research. *Sustainability* **2020**, *12*, 8950. [\[CrossRef\]](#)
98. Keathley-Herring, H.; Van Aken, E.; Gonzalez-Aleu, F.; Deschamps, F.; Letens, G.; Orlandini, P.C. Assessing the maturity of a research area: Bibliometric review and proposed framework. *Scientometrics* **2016**, *109*, 927–951. [\[CrossRef\]](#)
99. Zaccaria, V.; Rahman, M.; Aslanidou, I.; Kyprianidis, K. A Review of Information Fusion Methods for Gas Turbine Diagnostics. *Sustainability* **2019**, *11*, 6202. [\[CrossRef\]](#)
100. Shu, F.; Julien, C.-A.; Zhang, L.; Qiu, J.; Zhang, J.; Larivière, V. Comparing journal and paper level classifications of science. *J. Inf.* **2019**, *13*, 202–225. [\[CrossRef\]](#)
101. Leiva, M.A.; García, A.J.; Shakarian, P.; Simari, G.I. Argumentation-Based Query Answering under Uncertainty with Application to Cybersecurity. *Big Data Cogn. Comput.* **2022**, *6*, 91. [\[CrossRef\]](#)
102. Yang, L.; Meng, X.; Karniadakis, G.E. B-PINNs: Bayesian physics-informed neural networks for forward and inverse PDE problems with noisy data. *J. Comput. Phys.* **2021**, *425*, 109913. [\[CrossRef\]](#)
103. Goswami, S.; Anitescu, C.; Rabczuk, T. Adaptive fourth-order phase field analysis using deep energy minimization. *Theor. Appl. Fract. Mech.* **2020**, *107*, 102527. [\[CrossRef\]](#)
104. Costabal, F.S.; Yang, Y.; Perdikaris, P.; Hurtado, D.E.; Kuhl, E. Physics-Informed Neural Networks for Cardiac Activation Mapping. *Front. Phys.* **2020**, *8*, 42. [\[CrossRef\]](#)
105. Jagtap, A.D.; Mao, Z.; Adams, N.; Karniadakis, G.E. Physics-informed neural networks for inverse problems in supersonic flows. *arXiv* **2022**, arXiv:2202.11821.
106. Meng, X.; Li, Z.; Zhang, D.; Karniadakis, G.E. PPINN: Parareal physics-informed neural network for time-dependent PDEs. *Comput. Methods Appl. Mech. Eng.* **2020**, *370*, 113250. [\[CrossRef\]](#)



107. Haghighat, E.; Raissi, M.; Moure, A.; Gomez, H.; Juanes, R. A physics-informed deep learning framework for inversion and surrogate modeling in solid mechanics. *Comput. Methods Appl. Mech. Eng.* **2021**, *379*, 113741. [\[CrossRef\]](#)
108. Kharazmi, E.; Zhang, Z.; Karniadakis, G.E. hp-VPINNs: Variational physics-informed neural networks with domain decomposition. *Comput. Methods Appl. Mech. Eng.* **2021**, *374*, 113547. [\[CrossRef\]](#)
109. Fang, Y.; Wu, G.Z.; Wang, Y.Y.; Dai, C.Q. Data-driven femtosecond optical soliton excitations and parameters discovery of the high-order NLSE using the PINN. *Nonlinear Dyn.* **2021**, *105*, 603–616. [\[CrossRef\]](#)
110. Dourado, A.; Viana, F.A.C. Physics-Informed Neural Networks for Missing Physics Estimation in Cumulative Damage Models: A Case Study in Corrosion Fatigue. *J. Comput. Inf. Sci. Eng.* **2020**, *20*, 061007. [\[CrossRef\]](#)
111. Shin, Y.; Darbon, J.; Karniadakis, G.E. On the convergence of physics informed neural networks for linear second-order elliptic and parabolic type PDEs. *arXiv* **2020**, arXiv:2004.01806. [\[CrossRef\]](#)
112. Zobeiry, N.; Humfeld, K.D. A physics-informed machine learning approach for solving heat transfer equation in advanced manufacturing and engineering applications. *Eng. Appl. Artif. Intell.* **2021**, *101*, 104232. [\[CrossRef\]](#)
113. Mehta, P.P.; Pang, G.; Song, F.; Karniadakis, G.E. Discovering a universal variable-order fractional model for turbulent Couette flow using a physics-informed neural network. *Fract. Calc. Appl. Anal.* **2019**, *22*, 1675–1688. [\[CrossRef\]](#)
114. Liu, M.; Liang, L.; Sun, W. A generic physics-informed neural network-based constitutive model for soft biological tissues. *Comput. Methods Appl. Mech. Eng.* **2020**, *372*, 113402. [\[CrossRef\]](#)
115. Pu, J.; Li, J.; Chen, Y. Solving localized wave solutions of the derivative nonlinear Schrodinger equation using an improved PINN method. *arXiv* **2021**, arXiv:2101.08593. [\[CrossRef\]](#)
116. Meng, X.; Babaei, H.; Karniadakis, G.E. Multi-fidelity Bayesian neural networks: Algorithms and applications. *J. Comput. Phys.* **2021**, *438*, 110361. [\[CrossRef\]](#)
117. Jagtap, A.D.; Karniadakis, G.E. Extended physics-informed neural networks (XPINNs): A generalized space-time domain decomposition based deep learning framework for nonlinear partial differential equations. *Commun. Comput. Phys.* **2020**, *28*, 2002–2041. [\[CrossRef\]](#)
118. Pang, G.; D'Elia, M.; Parks, M.; Karniadakis, G. nPINNs: Nonlocal physics-informed neural networks for a parametrized nonlocal universal Laplacian operator. Algorithms and applications. *J. Comput. Phys.* **2020**, *422*, 109760. [\[CrossRef\]](#)
119. Rafiq, M.; Rafiq, G.; Choi, G.S. DSFA-PINN: Deep Spectral Feature Aggregation Physics Informed Neural Network. *IEEE Access* **2022**, *10*, 22247–22259. [\[CrossRef\]](#)
120. Raynaud, G.; Houde, S.; Gosselin, F.P. ModalPINN: An extension of physics-informed Neural Networks with enforced truncated Fourier decomposition for periodic flow reconstruction using a limited number of imperfect sensors. *J. Comput. Phys.* **2022**, *464*, 111271. [\[CrossRef\]](#)
121. Haitsiukevich, K.; Ilin, A. Improved Training of Physics-Informed Neural Networks with Model Ensembles. *arXiv* **2022**, arXiv:2204.05108.
122. Lahariya, M.; Karami, F.; Develder, C.; Crevecoeur, G. Physics-informed Recurrent Neural Networks for The Identification of a Generic Energy Buffer System. In Proceedings of the 2021 IEEE 10th Data Driven Control and Learning Systems Conference (DDCLS), Suzhou, China, 14–16 May 2021; pp. 1044–1049. [\[CrossRef\]](#)
123. Zhang, X.; Zhu, Y.; Wang, J.; Ju, L.; Qian, Y.; Ye, M.; Yang, J. GW-PINN: A deep learning algorithm for solving groundwater flow equations. *Adv. Water Resour.* **2022**, *165*, 104243. [\[CrossRef\]](#)
124. Yang, M.; Foster, J.T. Multi-output physics-informed neural networks for forward and inverse PDE problems with uncertainties. *Comput. Methods Appl. Mech. Eng.* **2022**, 115041. [\[CrossRef\]](#)
125. Psaros, A.F.; Kawaguchi, K.; Karniadakis, G.E. Meta-learning PINN loss functions. *J. Comput.* **2022**, *458*, 111121. [\[CrossRef\]](#)
126. Habib, A.; Yildirim, U. Developing a physics-informed and physics-penalized neural network model for preliminary design of multi-stage friction pendulum bearings. *Eng. Appl. Artif. Intell.* **2022**, *113*, 104953. [\[CrossRef\]](#)
127. Xiang, Z.; Peng, W.; Liu, X.; Yao, W. Self-adaptive loss balanced Physics-informed neural networks. *Neurocomputing* **2022**, *496*, 11–34. [\[CrossRef\]](#)
128. Page, M.J.; McKenzie, J.E.; Bossuyt, P.M.; Boutron, I.; Hoffmann, T.C.; Mulrow, C.D.; Shamseer, K.; Tetzlaff, J.M.; Akl, E.A.; Brennan, S.E.; et al. The PRISMA 2020 statement: An updated guideline for reporting systematic reviews. *BMJ* **2021**, *372*, n71. [\[CrossRef\]](#)