Hamiltonian neural network approach to fuzzball geodesics

Andrea Cipriani,[1,2,*] Alessandro De Santis,[2,3,4,†] Giorgio Di Russo,[2,5,‡] Alfredo Grillo,[1,2,§] and Luca Tabarroni[1,2,‖]

[1]*Dipartimento di Fisica, Università di Roma Tor Vergata, Via della Ricerca Scientifica 1, 00133 Roma, Italia*

[2]*Sezione INFN Roma 2, Via della Ricerca Scientifica 1, 00133 Roma, Italia*

[3]*Helmholtz-Institut Mainz, Johannes Gutenberg-Universität Mainz, 55099 Mainz, Germany*

[4]*GSI Helmholtz Centre for Heavy Ion Research, 64291 Darmstadt, Germany*

[5]*School of Fundamental Physics and Mathematical Sciences, Hangzhou Institute for Advanced Study, UCAS, Hangzhou 310024, China*

(Received 20 March 2025; accepted 16 June 2025; published 15 July 2025)

The recent increase in computational resources and data availability has led to a significant rise in the use of machine learning (ML) techniques for data analysis in physics. However, the application of ML methods to solve differential equations capable of describing even complex physical systems is not yet fully widespread in theoretical high-energy physics. Hamiltonian neural networks (HNNs) are tools that minimize a loss function defined to solve Hamilton equations of motion. In this work, we implement several HNNs trained to solve, with high accuracy, the Hamilton equations for a massless probe moving inside a smooth and horizonless geometry known as a D1-D5 circular fuzzball. We study both planar (equatorial) and nonplanar geodesics in different regimes according to the impact parameter, some of which are unstable. Our findings suggest that HNNs could eventually replace standard numerical integrators, as they are equally accurate but more reliable in critical situations.

DOI: 10.1103/dssv-x49b

I. INTRODUCTION

Physics-informed neural networks (PINNs) are a widely used tool in today's machine learning (ML) landscape. They consist of neural networks (NNs) that, during the training phase, learn to solve the differential equations governing the physical laws of a system in a model-independent way. When these differential equations correspond to Hamilton equations of motion, we refer to them as Hamiltonian neural networks (HNNs). The HNN paradigm was introduced in [1], and in the present work we closely follow the strategy proposed in [2]. The key advantages of HNNs over standard numerical integrators can be summarized as follows:

(1) The predicted solution is analytical in time and not limited to a discrete set of time steps.
(2) Conservation laws, symmetries, constraints, and prior knowledge of the system can be easily incorporated at the level of the architecture and of the loss function to improve the predictability of the HNN.
(3) The minimization process of the loss function occurs under the constraint that the solution must satisfy the system of equations at all times simultaneously and independently, thus avoiding any iterative mechanism.

The last point is crucial for systems whose (effective) potential exhibits unstable critical points, for which small changes of the initial conditions lead to completely different behaviors of the solutions, similar to what happens in chaotic dynamical systems. As is well known, any numerical integrator operating iteratively, building the solution based on the result of the previous time step, accumulates errors and inevitably loses accuracy over long timescales in such situations. HNNs have the potential to overcome this issue.

While HNNs have been applied in various physical contexts,[1] their diffusion in high-energy physics remains limited. In this paper, we apply this technique for the first time in the field of string theory, where ML has only begun

*Contact author: andrea.cipriani@roma2.infn.it
†Contact author: desantia@uni-mainz.de
‡Contact author: gdr794@ucas.ac.cn
§Contact author: alfredo.grillo89@gmail.com
‖Contact author: luca.tabarroni@roma2.infn.it

[1]Due to the rapidly increasing number of ML applications, the literature on this topic is vast, and providing a comprehensive review is beyond the scope of this article. An incomplete selection of applications includes [3–17].

to gain attraction in recent years [18–36]. Specifically, we focus on using HNNs to determine the noncritical and unstable critical geodesics of a massless particle moving in a D1-D5 circular fuzzball geometry by solving the associated equations of motion. The scattering properties, scalar perturbations, and tidal deformations of this physical system have already been extensively studied in [37–39]. It serves as a well-suited test case, given that the exact trajectory, to which we will refer as the *ground truth*, can be explicitly computed. Thus, the results presented in this paper do not add new physical insights into the system under study; rather, the work is methodological in nature and aims to assess the feasibility and performance of HNNs compared to standard numerical methods in a controlled setting. Our investigation, as clearly demonstrated by the numerical results, shows that the HNN performs comparably to numerical integrators for noncritical trajectories and surpasses them for unstable critical trajectories, predicting accurate solutions even over long timescales.

This result suggests that HNNs, and PINNs in general, may offer a robust and reliable solution for numerically determining unstable solutions, not only in string theory, but in all systems where the underlying physics gives rise to complex dynamics. This one, together with other features discussed later on, is one of the reasons that favor these machine-learning-based algorithms over standard integrators, even when their performances appear comparable. Moreover, the promising outcomes of this paper pave the way for a systematic study of more intricate geometries, where the absence of a ground truth or the lack of physical information necessitates reliable and flexible tools for analysis. HNNs and PINNs are ideal candidates in this regard. As an immediate follow-up, an interesting direction is the study of the so-called D1-D5-P (three-charge) fuzzball, where, due to its intrinsic complexity, an analysis in terms of either geodesic motion or wave propagation has never been completed in full generality. Such a study would also enable a subsequent analysis of the quasinormal modes (QNMs) spectrum [40–42], which constitutes one of our long-term objectives.

This paper is organized as follows: In Sec. II, we introduce the HNN-based strategy for solving a generic set of Hamiltonian equations. In Sec. III, we review the theoretical background of the D1-D5 fuzzball and its geodesics. In Sec. IV, we present numerical results for planar geodesics, while results for the nonplanar ones are discussed in Sec. V. Additionally, this work is accompanied by several Appendixes providing supplemental material to the main text.

II. HAMILTONIAN NEURAL NETWORKS

In this section, we provide an introduction to HNNs by summarizing the basic concepts. We refer mostly to Refs. [1,2], from which we have taken inspiration for this work.

A. Hamiltonian mechanics

The phase space in which the Hamiltonian formalism takes place is defined by the generalized space coordinates $\mathbf{q} = (q^1, \ldots, q^n)$ and by the generalized conjugate momenta $\mathbf{p} = (p_1, \ldots, p_n)$, where $n \geq 1$ denotes the number of degrees of freedom. Starting from an initial state at time t_0, the evolution of the mechanical system to a final state at t_1 is described by

$$(\mathbf{q}_1, \mathbf{p}_1) = (\mathbf{q}_0, \mathbf{p}_0) + \int_{t_0}^{t_1} \mathbf{S}(\mathbf{q}, \mathbf{p}) \mathrm{d}t, \tag{2.1}$$

where

$$\mathbf{S}(\mathbf{q}, \mathbf{p}) = \left(\frac{\mathrm{d}\mathbf{q}}{\mathrm{d}t}, \frac{\mathrm{d}\mathbf{p}}{\mathrm{d}t} \right) \tag{2.2}$$

is the time derivatives vector. The core of Hamiltonian mechanics is the definition of a scalar function $\mathcal{H}(\mathbf{q}, \mathbf{p})$ called a Hamiltonian. This function is defined in such a way that the Hamilton equations are satisfied[2]:

$$\dot{\mathbf{q}} = \frac{\partial \mathcal{H}}{\partial \mathbf{p}}, \qquad \dot{\mathbf{p}} = -\frac{\partial \mathcal{H}}{\partial \mathbf{q}}. \tag{2.3}$$

This allows us to define the *symplectic gradient* from Eq. (2.2) as

$$\mathbf{S}_{\mathcal{H}} = \left(\frac{\partial \mathcal{H}}{\partial \mathbf{p}}, -\frac{\partial \mathcal{H}}{\partial \mathbf{q}} \right) \tag{2.4}$$

and therefore, thanks to Eq. (2.1), we can evaluate the evolution of the system. Furthermore, since along the vector $\mathbf{S}_{\mathcal{H}}$ the Hamiltonian is conserved, we can define the conserved quantity $\mathrm{E} = \mathcal{H}(\mathbf{q}, \mathbf{p})$. In the following, we will occasionally use the definition of the vector $\mathbf{z} \in \mathbb{R}^{2n}$ as

$$\mathbf{z} = (\mathbf{q}, \mathbf{p})^T. \tag{2.5}$$

B. HNNs

Now, we present the strategy based on Hamiltonian neural networks (HNNs) to solve Eq. (2.3) for a generic system. In this work, we will then specialize to the case of the D1-D5 fuzzball geometry, whose details are provided in Sec. III. In order to start appreciating the differences between machine learning and standard numerical integrators, let us recall the widely used first-order semi-implicit Euler method to evaluate a trajectory in the time range $[0, T]$. This method respects the symplectic structure of the Hamiltonian system. In particular, it automatically conserves a quantity which is the original Hamiltonian shifted

[2]The notation $\dot{x}$ denotes the time derivative.

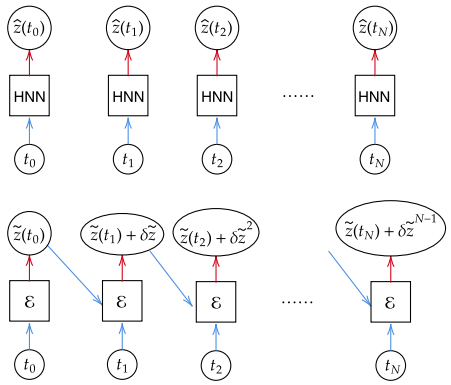

FIG. 1. Representation of the algorithmic flow for a Hamiltonian neural network (HNN, on top) and for the first-order semi-implicit Euler method ($\mathcal{E}$, on bottom). Blue arrows refer to the inputs at each step, while the red ones refer to the outputs. As can be seen, the former produces an output individually for each time without using the information at the previous step, while the latter works iteratively, accumulating errors.

by a term proportional to the discretization time step. In fact, the algorithm requires splitting the time interval into $N+1$ points equally spaced with time step Δt such that $T = N\Delta t$. The solution is then computed in correspondence with the discrete times $t_j = j\Delta t$, with $j = 0, \ldots, N$, by updating the generalized coordinates and the corresponding momenta[3] at time step $j+1$ from those at time step j:

$$\tilde{q}_k(t_{j+1}) = \tilde{q}_k(t_j) + \Delta t \frac{\partial \mathcal{H}(\tilde{\mathbf{q}}(t_j), \tilde{\mathbf{p}}(t_j))}{\partial p_k},$$
$$\tilde{p}_k(t_{j+1}) = \tilde{p}_k(t_j) - \Delta t \frac{\partial \mathcal{H}(\tilde{\mathbf{q}}(t_{j+1}), \tilde{\mathbf{p}}(t_j))}{\partial q_k}. \tag{2.6}$$

By employing this iterative method, it is inevitable to accumulate errors through all the input time series, usually causing the prediction of trajectories, especially at large times, to be affected by large discretization errors. In practice, for particularly complex systems, accurate solutions are found only in the limit of $\Delta t \mapsto 0$ (i.e., $N \mapsto \infty$), where the method is expected to preserve the real Hamiltonian. On the contrary, a HNN provides an analytical solution independently for any input time. This methodological difference is schematically sketched in Fig. 1.

Recalling the definition of Eq. (2.5), we can write the HNN solution as[4]

$$\hat{\mathbf{z}}(t, \mathbf{w}) = \mathbf{z}_0 + f(t)\mathbf{O}(t, \mathbf{w}), \tag{2.7}$$

[3]The notation $\tilde{x}$ denotes a quantity obtained with a numerical integrator.

[4]The notation $\hat{x}$ denotes a quantity predicted by a neural network.

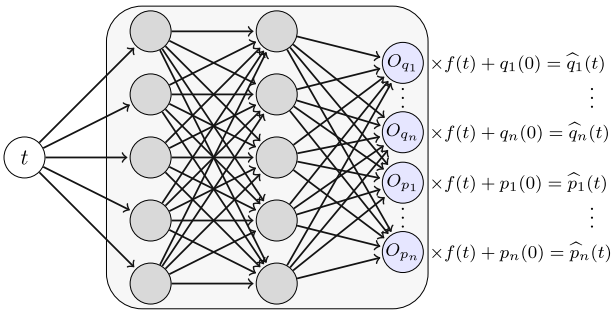

FIG. 2. The neural network architecture consists of one input node for the time t, an arbitrary number of hidden layers, and an output layer with a number of nodes (colored light blue) equal to the number of nontrivial Hamilton equations that constitute the system. The network outputs are multiplied by $f(t) = 1 - e^{-t}$ and added to the initial conditions. Notice that $f(0) = 0$ is such that the initial conditions are exactly satisfied by the predicted solution.

where

$$\mathbf{O}(t, \mathbf{w}) = (O_{q_1}(t, \mathbf{w}), \ldots, O_{q_n}(t, \mathbf{w}), O_{p_1}(t, \mathbf{w}), \ldots, O_{p_n}(t, \mathbf{w}))^T \tag{2.8}$$

is the output of the NN whose architecture is designed to be a $\mathbb{R} \mapsto \mathbb{R}^{2n}$ map, with the time t being the input, and $\mathbf{z}_0 \in \mathbb{R}^{2n}$ is the vector of the initial conditions[5] at $t = 0$. The functions O_{q_k} and O_{p_k}, with $k = 1, \ldots, n$, represent single output nodes assuming values in $\mathbb{R}$. The single variables in Eq. (2.7) are therefore obtained according to

$$\hat{q}_k(t, \mathbf{w}) = q_k(0) + f(t)O_{q_k}(t, \mathbf{w}),$$
$$\hat{p}_k(t, \mathbf{w}) = p_k(0) + f(t)O_{p_k}(t, \mathbf{w}). \tag{2.9}$$

See also Fig. 2 to visualize the network diagram. The symbol $\mathbf{w}$ represents the collection of the network weights. We will specify the network architecture in the results section, since it is not necessary for the purpose of explaining the method. Let us only clarify that the choice of the network architecture has to be such that the expression $\hat{\mathbf{z}}(t, \mathbf{w})$ is analytical and differentiable with respect to the time for any set of weights $\mathbf{w}$. We explicitly write in this section the symbol $\mathbf{w}$ for the sake of clarity, but we will omit it further on in the text, leaving it understood that whenever a variable is marked by the symbol $\hat{}$, it will be dependent on the architecture and weights. The function $f(t)$, which does not contain trainable parameters, is chosen in such a way that it satisfies $f(0) = 0$, and it is introduced

[5]In the case in which some momenta are conserved, due to the fact that the corresponding coordinates are cyclic, then we will not specify the nodes associated with them. This is exactly what we mean by "nontrivial Hamilton equations" in the caption of Fig. 2.

to automatically enforce the initial conditions.[6] In this work, we make the same choice as in Ref. [2] [where the impact of different choices of $f(t)$ is also illustrated] and define the bounded function

$$f(t) = 1 - e^{-t}. \tag{2.10}$$

We now explain in detail how the solutions to Eq. (2.3), parametrized by (2.9), can be obtained through the minimization of a specifically designed loss function. Let us consider again the time interval $[0, T]$ sampled with $N+1$ points at constant time step Δt and indicate by $t_j = j\Delta t$ the time after j steps. We define the loss function

$$L_{\rm dyn}(\mathbf{w}) = \sum_{k=1}^{n}\left\{\frac{1}{N}\sum_{j=0}^{N}\gamma_{q_k}\left[\left(\dot{\hat{q}}_k - \widehat{\frac{\partial \mathcal{H}}{\partial p_k}}\right)\Big|_{t_j}\right]^2 + \frac{1}{N}\sum_{j=0}^{N}\gamma_{p_k}\left[\left(\dot{\hat{p}}_k + \widehat{\frac{\partial \mathcal{H}}{\partial q_k}}\right)\Big|_{t_j}\right]^2\right\}, \tag{2.11}$$

where

$$\dot{\hat{q}}_k|_{t_j} = \left.\frac{{\rm d}f(t)}{{\rm d}t}\right|_{t_j}\cdot O_{q_k}(t_j, \mathbf{w}) + f(t_j)\cdot\left.\frac{{\rm d}O_{q_k}(t, \mathbf{w})}{{\rm d}t}\right|_{t_j}, \qquad \left.\widehat{\frac{\partial \mathcal{H}}{\partial p_k}}\right|_{t_j} = \left.\frac{\partial \mathcal{H}}{\partial p_k}\right|_{\hat{\mathbf{z}}(t_j, \mathbf{w})}. \tag{2.12}$$

Similarly, one can obtain the definitions for $\dot{\hat{p}}_k|_{t_j}$ and $\widehat{\frac{\partial \mathcal{H}}{\partial q_k}}|_{t_j}$ in Eq. (2.11) by replacing $\hat{q}_k$ with $\hat{p}_k$ and vice versa. The factors γ_{q_k} and γ_{p_k} can be chosen to weight a variable more or less, relative to the others. Throughout most of the work, we will simply use $\gamma_{q_k} = \gamma_{p_k} = 1$.

Equation (2.12) makes manifest the dependence of $L_{\rm dyn}(\mathbf{w})$ on the network weights and on the way $f(t)$ enters the loss function. The subscript "dyn" refers to the fact that, for a set of weights such that $L_{\rm dyn}(\mathbf{w}) = 0$, the network predictions of Eq. (2.9) would exactly satisfy the dynamics expressed by the Hamilton equations (2.3) at the discrete times t_j. The possibility to actually meet the condition $L_{\rm dyn}(\mathbf{w}) = 0$ is supported by the so-called universal approximation theorem [43,44], which states that a large enough neural network will be able to approximate any continuous function with arbitrary accuracy. In practice, this possibility is spoiled by the fact that a neural network has a finite number of neurons and then, as is customary in the machine learning field, we can at most search for the set of weights that make the loss function as small as possible—i.e., we want to solve the equation

$$\frac{\partial L_{\rm dyn}(\mathbf{w})}{\partial \mathbf{w}} = 0 \tag{2.13}$$

and find an approximation to the solution. We will call *training* the process of minimization of the loss function, but it has to be emphasized that, unlike common regression problems, the training is not data-driven but rather equation-driven. This means that the only information that will allow the NN to fulfill the desired task is codified in the Hamilton equations, which are analytically known after defining the Hamiltonian, and no assumption on the true trajectory is required. From this perspective, the HNNs alone offer a model-independent solution to the problem, and additional information—e.g., the knowledge of the solution at given times due to experiments or different methods—can be easily incorporated in the training process to improve the accuracy of the predicted solution.

Let us stress that the formulas given in Eq. (2.9) are continuous and differentiable in the variable t; then, after the training, they represent an *analytical* solution to the Hamilton equations, providing a significant advantage over numerical integrators, where the solution is only known at predefined discrete time points. However, a natural concern is the reliability of the prediction for times different from the t_j entering $L_{\rm dyn}$. In addition to regression tasks, HNNs can suffer from overfitting by providing an accurate solution for $t = t_j$, but one considerably different from the ground truth for t in between t_j and t_{j+1}. A demonstrative example of the problem is shown in Fig. 3. This problem is expected to be particularly severe as Δt increases ($N \to 0$), since the solution is less constrained in between two consecutive points, and it vanishes in the limit $\Delta t \to 0$ ($N \to \infty$). One possibility to overcome the problem is to sample the interval $[0, T]$ differently at the beginning of each epoch, for instance, by adding Gaussian noise to a given initial set of times t_j, or to choose a large N value and check *a posteriori* that the loss function, computed for a set of times different from t_j, keeps at the same level as the one minimized during the training. In this work, we have followed the latter strategy. Of course, there is no chance that the prediction could be

[6]We remark that this might also be achieved by explicitly including the conditions at the level of the loss function. The advantage of the definitions in Eq. (2.9) is to automatically restrict the space of possible solutions by accelerating and stabilizing the minimization of the loss function.

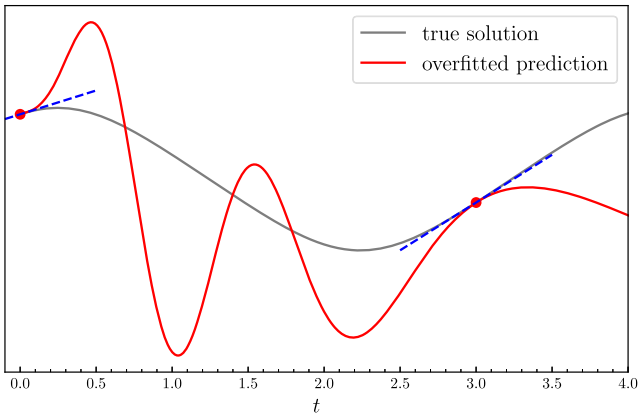

FIG. 3. Demonstrative example of overfitting for a HNN. In this case, the dynamical loss is minimized only for $t = 0$ and $t = 3$ (red points), in correspondence with which the predicted solution is accurate, while it rapidly deviates from the true one otherwise.

accurate for t outside the interval $[0, T]$, since the network has not been instructed on the dynamics in that regime.

Another great advantage of the procedure outlined above is the possibility to extend the definition of the loss function by including terms accounting for the conservation of physical quantities. Particularly important is the term imposing the energy conservation, already introduced in Ref. [2], that can be defined as

$$L_{\text{energy}}(\mathbf{w}) = \frac{1}{N}\sum_{j=0}^{N}[\mathcal{H}(\hat{\mathbf{z}}(t_j, \mathbf{w})) - E_0]^2, \tag{2.14}$$

where $E_0 = \mathcal{H}(\mathbf{z}_0)$ is the value of the energy to be conserved along the trajectory. We then define the total loss function that has to be minimized as a convex linear combination of L_{dyn} and L_{energy},

$$L(\lambda, \mathbf{w}) = (1-\lambda)L_{\text{dyn}}(\mathbf{w}) + \lambda L_{\text{energy}}(\mathbf{w}), \tag{2.15}$$

where we have introduced the trade-off parameter $\lambda \in [0, 1]$. The practical advantage of introducing the energy conservation term in the loss function is making the convergence toward the correct solution faster and the training procedure more stable. Moreover, setting $\lambda > 0$ serves also as a regulator to select a unique solution in those situations where L_{dyn} might admit multiple ones. We will show, in fact, that setting $\lambda > 0$ dramatically affects the performance of the neural network in some cases. Similarly, one can introduce other terms enforcing the predicted trajectories to conserve physical quantities.

However, the loss function of Eq. (2.15) is not particularly helpful in comparing the performances of the neural network at varying λ, since λ itself, especially when close to 0 or 1, alters the order of magnitude of $L(\lambda, \mathbf{w})$ at a given $\mathbf{w}$. During the training, we therefore also monitor individually $L_{\text{dyn}}(\mathbf{w})$ and $L_{\text{energy}}(\mathbf{w})$ in order to have a measure of the goodness of the solution independently from λ.

Further regularization terms limiting the space of the solutions according to additional knowledge about the system can be introduced as well to improve the accuracy of the network. We will come back to it later in the results section, since this is case dependent.

We conclude this section by observing that we have specialized Eq. (2.11) for the case of the Hamilton equations, but in general one can write a proper loss function to search for the solution of any system of ordinary differential equations. For instance, the geometry described in the next section also enjoys the separability of the motion, and it allows the derivation of a different, but physically equivalent, set of equations of motion. We will consider this alternative point of view in Appendix E, with the aim of highlighting differences between the usage of HNNs. The scope of applicability of PINNs is therefore extremely wide, and such technology is a valuable tool that can complement standard numerical integrators and, at times, replace them.

In the next section, we will introduce the physical system whose equations of motion are the ones we want to solve by implementing a HNN.

III. D1-D5 FUZZBALL GEOMETRY

The physical configuration we consider in this work is the motion of a scalar massless neutral particle inside the geometry of a D1-D5 circular fuzzball. This section is divided into the following subsections: in Sec. III A, we furnish a presentation of the background in which the fuzzball comes out, with the aim of understanding its nature and why it is playing an interesting role in string theory; in Sec. III B, we recall the basics for determining the Hamiltonian of a particle that is moving inside a generic spacetime; in Sec. III C, we focus on the aforementioned case of D1-D5 fuzzball geometry, writing the Hamiltonian of the particle and presenting some relevant properties of such geometry; and in Secs. III D and III E, we distinguish the two cases of interest: planar and nonplanar motion. Details concerning the formulas that will be shown can be found in the Appendixes.

A. Presentation of the D1-D5 fuzzball

As already said, in this paper we study the geodesic motion (i.e., the free-fall trajectory) of a massless particle moving inside a specific geometry derived from a low-energy limit of string theory, exploring it as another potential application of the powerful and versatile techniques of ML in this field. To grasp the physical context, we give a brief review of the concept of black holes (BHs) and its relevance in modern theoretical physics. These gravitational objects, first discovered within the context of general relativity, exhibit several peculiar properties.

They are surrounded by a surface called the event horizon, beyond which nothing, not even light, can escape—hence, the term "black" in their name. Furthermore, at the core of the BH lies a singularity, known as a curvature singularity, where spacetime is no longer well defined. The cause is the fact that general relativity becomes unreliable at scales of the energy of the singularity, and quantum effects should be properly taken into account.

As a matter of fact, BHs are an interesting puzzle for theoretical physicists, and this is due to their simple nature: they are described by a handful of parameters—namely, their mass, charge, and angular momentum—while being the result of the gravitational collapse of very complex physical systems, such as stars. Moreover, no matter the complexity of the quantum state that crosses the event horizon, this only results in a change of those parameters. This is typically referred to as the *information-loss paradox* [45] and reveals a curious—albeit not yet understood—relationship between general relativity and quantum mechanics. Remaining still at the classical level, it was shown that BHs are akin to thermodynamical systems: they obey laws similar to those of thermodynamics [46] and possess an associated entropy-like quantity in the form of the event horizon's area. It is well understood that entropy has a microscopic interpretation in terms of the number of microstates, a description that is woefully absent in general relativity, since here it is usually said that "black holes have no hair" (meaning, as said before, that they are described only by mass, charge, and angular momentum).

As it is, both the existence of a singularity and that of the event horizon call for a theory of quantum gravity and a better understanding of the classical BHs. In the context of string theory, a proposal can be formulated (the "fuzzball proposal" [47–49]), according to which BHs should be thought of as superpositions of quantum states, some of which admit a classical description as smooth and horizonless gravity solutions with the same mass, charge, and angular momentum as the BH. In addition, these microstates can emit radiation, preserving quantum information and hence solving the information-loss paradox.

There are many ways to construct these smooth, horizonless geometries, and one of these is to consider particular objects naturally arising in string theory, the Dp-branes. Without entering into too much detail, (super) string theory can be defined in a mathematically consistent way only in $9+1$ spacetime dimensions; Dp-branes extend both along the time direction of spacetime and along p spatial dimensions. They carry mass and charge, and consequently are subjected to the action of forces. One can combine a number of different branes in order to form a stable gravitational object, with a given mass and charge. For the scope of this work, we are interested in the gravitational object—and its associated spacetime metric—which is obtained by combining D1 and D5 branes, and which has a circular profile.

B. Formalism

Given a spacetime of dimension d, it is well known that the scalar product between any two of its vectors is codified by a symmetric $d \times d$ matrix, which is the metric $g_{\mu\nu}$. In this way, the squared length of the line element is given by

$$\mathrm{d}s^2 = g_{\mu\nu}\mathrm{d}x^\mu \mathrm{d}x^\nu, \tag{3.1}$$

where x^μ are the spacetime coordinates ($\mu = 0, \ldots, d-1$). The free fall of a neutral particle inside a certain spacetime with metric $g_{\mu\nu}$ happens along the geodesics, which are solutions of differential equations known as *geodesic equations*. On the other hand, it is also known that the trajectories in the physical space are such that the action is minimized, which can be done by solving the *Euler-Lagrange equations*

$$\frac{\mathrm{d}}{ds}\frac{\partial \mathcal{L}}{\partial \dot{x}^\mu} = \frac{\partial \mathcal{L}}{\partial x^\mu}, \tag{3.2}$$

where s is the affine parameter of the geodesic of the particle and $\mathcal{L}$ is its Lagrangian. From all of this, we understand that the geodesic equations are equivalent to the Euler-Lagrange ones, and it can be shown that it is possible if the Lagrangian has the following form:

$$\mathcal{L} = \frac{1}{2} g_{\mu\nu}\dot{x}^\mu \dot{x}^\nu, \qquad \dot{x}^\mu = \frac{\mathrm{d}x^\mu}{\mathrm{d}s}. \tag{3.3}$$

In order to pass to the Hamiltonian viewpoint, we have to compute the canonical conjugate momenta

$$P_\mu = \frac{\partial \mathcal{L}}{\partial \dot{x}^\mu}, \tag{3.4}$$

and the Hamiltonian $\mathcal{H}$ is obtained through a Legendre transformation of the Lagrangian

$$\mathcal{H} = P_\mu \dot{x}^\mu(P) - \mathcal{L}(x, \dot{x}(P)) = \frac{1}{2} g^{\mu\nu} P_\mu P_\nu. \tag{3.5}$$

Notice that what we have called here x and P correspond to $\mathbf{q}$ and $\mathbf{p}$ of Sec. II A, but this is the typical notation in the gravity context. In addition, we consider as time the affine parameter s in place of the coordinate time t used in Sec. II B. This implies no conceptual and practical differences in the ML strategy proposed. The mass shell condition for a particle of mass μ_0 reads [37,50] $\mathcal{H} = -\frac{1}{2}\mu_0^2$. Since we are interested in massless geodesics, we obtain the following important condition: $\mathcal{H} = 0$. Finally, we recall that, if the metric components—and thus, the Lagrangian—do not depend on a specific coordinate x^μ (which is said to be cyclic), then the corresponding momentum P_μ is conserved, as can be read from the Euler-Lagrange equations (3.2).

C. Geometry

Now, we pass to examine the geometry of the D1-D5 circular fuzzball and the properties of the motion of a massless particle inside this. The spacetime metric generated by this object is

$$\begin{aligned} ds^2 &= H^{-1}[-(dt+\omega_\phi d\phi)^2+(dz+\omega_\psi d\psi)^2] \\ &+ H\left[(\rho^2+a_f^2\cos^2\theta)\left(\frac{d\rho^2}{\rho^2+a_f^2}+d\theta^2\right)\right. \\ &\left.+\rho^2\cos^2\theta d\psi^2+(\rho^2+a_f^2)\sin^2\theta d\phi^2\right] \\ &+\left(\frac{H_1}{H_5}\right)^{1/2}dz_{1,2}^2+\left(\frac{H_5}{H_1}\right)^{1/2}dz_{3,4}^2, \end{aligned} \tag{3.6}$$

where

$$\begin{aligned} &H=\sqrt{H_1H_5}, \qquad H_i=1+\frac{L_i^2}{\rho^2+a_f^2\cos^2\theta}, \\ &\omega_\phi=\frac{a_fL_1L_5\sin^2\theta}{\rho^2+a_f^2\cos^2\theta}, \qquad \omega_\psi=\frac{a_fL_1L_5\cos^2\theta}{\rho^2+a_f^2\cos^2\theta}. \end{aligned} \tag{3.7}$$

Here, a_f is the radius of the circular profile, and L_1 and L_5 are the charges of, respectively, the D1 and D5 branes.[7] These are the parameters of the geometry that we are considering, and it is also rotating. It has to be said that the complete solution describing the fuzzball does not contain only the metric (3.6), but also other fields, which are not relevant for the dynamics of the neutral probe we are considering, and for this reason we have not written them. This spacetime is ten-dimensional,[8] but four out of the ten coordinates, z_I with $I=1,\ldots,4$, parametrize compact directions, and for this reason we can safely set their corresponding momenta P_{z_I} equal to 0 and neglect these coordinates. The coordinates (ρ,θ,ϕ,ψ) are called *oblate spheroidal coordinates*, and they are related to the standard Cartesian ones (x_1,x_2,x_3,x_4) through the relations

$$\begin{aligned} x_1&=\sqrt{\rho^2+a_f^2}\sin\theta\cos\phi, \\ x_2&=\sqrt{\rho^2+a_f^2}\sin\theta\sin\phi, \\ x_3&=\rho\cos\theta\cos\psi, \\ x_4&=\rho\cos\theta\sin\psi. \end{aligned} \tag{3.8}$$

[7]Notice that L_1 and L_5 have the dimension of a length, while in measure units with $c=G=\frac{1}{4\pi\epsilon_0}=1$, the charge has the dimension of a squared length. Consequently, the real charge is related to L_i via multiplication by a dimensional parameter.

[8]For $L_1=L_5=0$ (i.e., $H_1=H_5=1$ and $\omega_\phi=\omega_\psi=0$) and under a suitable coordinate change, the metric (3.6) describes flat spacetime.

Notice that the metric components do not depend on t, ϕ, ψ and z; consequently, the associated momenta $P_t=-E$, $P_\phi=J_\phi$, $P_\psi=J_\psi$, and P_z are conserved. Following Eqs. (3.3)–(3.5), we can arrive at the expression of the Hamiltonian of a particle moving inside this geometry, which is

$$\begin{aligned} \mathcal{H}=\frac{1}{2(\rho^2+a_f^2\cos^2\theta)H}&\left[P_\rho^2(\rho^2+a_f^2)+P_\theta^2\right. \\ &+\frac{(J_\psi a_f-P_zL_1L_5)^2}{\rho^2}+\frac{J_\psi^2}{\cos^2\theta}+\frac{J_\phi^2}{\sin^2\theta} \\ &-\frac{(J_\phi a_f-EL_1L_5)^2}{\rho^2+a_f^2}-(E^2-P_z^2)(\rho^2+a_f^2+L_1^2+L_5^2) \\ &\left.+(E^2-P_z^2)a_f^2\sin^2\theta\right]. \end{aligned} \tag{3.9}$$

From this, we can compute the derivatives, as required by Hamilton equations, that should then be used inside the loss function as explained in the previous Sec. II. We will not report these derivatives for the most general case, since they are not important for our study; we write in Appendix A the derivatives that are related to the *planar* and specific *nonplanar* motions, as we will soon explain.

Before proceeding with this, we have to pay attention to a particular feature of this geometry, and that will constitute another aspect to probe the goodness of the HNN—i.e., the *separability* of the dynamics along the radial ρ and the angular θ directions. Indeed, from the previous expression of the Hamiltonian, since it has to be 0 for the massless particle, we can neglect the overall factor and, by examining the terms within the square brackets, we notice that, by introducing the *Carter constant* K^2 [51], we can distinguish the motion along the direction parametrized by ρ from the one along the direction parametrized by θ. In particular,

$$\begin{aligned} P_\rho^2=Q_R(\rho)=(E^2-P_z^2)&\left(1+\frac{L_1^2+L_5^2}{\rho^2+a_f^2}\right) \\ &+\frac{(J_\phi a_f-EL_1L_5)^2}{(\rho^2+a_f^2)^2}-\frac{(J_\psi a_f-P_zL_1L_5)^2}{\rho^2(\rho^2+a_f^2)}-\frac{K^2}{\rho^2+a_f^2} \end{aligned} \tag{3.10}$$

and

$$P_\theta^2=Q_A(\theta)=K^2-\frac{J_\psi^2}{\cos^2\theta}-\frac{J_\phi^2}{\sin^2\theta}-(E^2-P_z^2)a_f^2\sin^2\theta. \tag{3.11}$$

Thanks to this, it is possible to integrate these two equations, getting expressions in terms of elliptic integrals. We are not going to do this in the most general case, since we are interested in the planar and a specific nonplanar

motions. In the former situation, we report also the procedure that shows how to integrate these equations, getting the ground truth that will constitute the reference point with respect to which we compare the results coming from the HNN and the standard numerical integrator.

A common peculiarity of BHs, D-branes, and other compact gravitating objects is the presence of "photon spheres" or light rings (light halos for rotating objects), which are stable or unstable bound orbits of particles separating the asymptotically flat region from the horizon (or the inner region for fuzzballs) [52–54]. In the following, we will denote them as *critical geodesics*. The simultaneous vanishing of the radial momentum $P_\rho^2(\rho) \equiv Q_R(\rho)$ and its derivative

$$Q_R(\rho_c, J_c) = Q_R'(\rho_c, J_c) = 0 \tag{3.12}$$

individuate the critical radius ρ_c and the corresponding critical angular momentum J_c. The radial effective potential $-Q_R(\rho)$ depends also on all the other parameters of the geometry—hence, fixing all of them, we get a particular shape, corresponding to which ρ_c can correspond either to a maximum or to a minimum. In the former case, the critical geodesic is unstable; in the latter, it is stable. It is worth noticing that this identification is possible only if the geometry is separable. Determining the critical geodesics for nonseparable geometries is something not clear, and HNNs could help in this direction.

We pass now to examine the planar and nonplanar cases separately, determining also the critical geodesics.

D. Planar case

Due to the axial (non–fully spherical) symmetry, the geodesics are nonplanar in general. Some simplifications arise for geodesics when the motion exists in a fixed plane. In order to find this, we should solve the Euler-Lagrange equation for the θ coordinate (which is a second-order differential equation), imposing as initial conditions $\theta(s_0) = \theta_0$ (the value of the angle corresponding to which we have the plane) and $\dot\theta(s_0) = 0$, where s_0 is the value of the affine parameter at the beginning of the motion. In this way, we obtain a Cauchy problem, and if the only solution is $\theta(s) = \theta_0$, then the motion takes place only along the plane corresponding to $\theta = \theta_0$.

For the geometry we are studying, this happens if $\theta_0 = 0, \pi/2$, which corresponds to the two equatorial planes.[9] Since θ is constant, $P_\theta = 0$. Furthermore, in order to avoid divergences in Eq. (3.11) for $\theta = \pi/2$, we are forced to take $J_\psi = 0$; conversely, $J_\phi = 0$ for $\theta = 0$. Furthermore, we also set $P_z = 0$, which simplifies the treatment without altering the final results. In the following, we will consider only the case with $\theta_0 = \pi/2$. The Hamiltonian (3.9) becomes

$$\begin{aligned}\mathcal{H} = \frac{1}{2\rho^2 H(\theta=\pi/2)}\Bigg[&P_\rho^2(\rho^2+a_f^2) + J_\phi^2 \\ &+ -\frac{(J_\phi a_f - E L_1 L_5)^2}{\rho^2+a_f^2} - E^2(\rho^2 + a_f^2 + L_1^2 + L_5^2) \\ &+ E^2 a_f^2\Bigg].\end{aligned} \tag{3.13}$$

Its derivatives are reported in Appendix A.

For what concerns the separability, Eq. (3.11) with $P_\theta = 0$, $\theta = \pi/2$, $J_\psi = 0$, $P_z = 0$ can be solved for the Carter constant

$$K^2 = J_\phi^2 + E^2 a_f^2, \tag{3.14}$$

which, when plugged into the radial equation (3.10), gives

$$P_\rho^2 = E^2\left(1 + \frac{L_1^2 + L_5^2}{\rho^2 + a_f^2}\right) + \frac{(J_\phi a_f - E L_1 L_5)^2}{(\rho^2+a_f^2)^2} - \frac{J_\phi^2 + a_f^2 E^2}{\rho^2 + a_f^2}. \tag{3.15}$$

We introduce now the impact parameter b:

$$b = \frac{J_\phi}{E}. \tag{3.16}$$

In this way, these equations can be expressed in terms of such parameters. Using the expressions of the momenta P_i in terms of the derivatives of the coordinates, $\dot x^i$, from Eq. (3.4) and inverting all these relations, we can find an equation for the derivative of ρ and ϕ with respect to the coordinate time t. At the end, we get

$$\begin{aligned}\frac{\mathrm{d}\rho(t)}{\mathrm{d}t} &= \frac{\dot\rho}{\dot t} = \pm\frac{(\rho^2+a_f^2)\sqrt{L_1^2(L_5^2+\rho^2) + \rho^2(L_5^2+\rho^2-b^2) + a_f^2(L_1^2+L_5^2+\rho^2) - 2a_f b L_1 L_5}}{(L_1^2+\rho^2)(L_5^2+\rho^2) + a_f^2(L_1^2+L_5^2+\rho^2) - a_f b L_1 L_5}, \\ \frac{\mathrm{d}\phi(t)}{\mathrm{d}t} &= \frac{\dot\phi}{\dot t} = \frac{b\rho^2 + a_f L_1 L_5}{(L_1^2+\rho^2)(L_5^2+\rho^2) + a_f^2(L_1^2+L_5^2+\rho^2) - a_f b L_1 L_5}.\end{aligned} \tag{3.17}$$

[9]Actually, these are specific cases of a more general class of geodesics, denoted as *shear-free* [38].

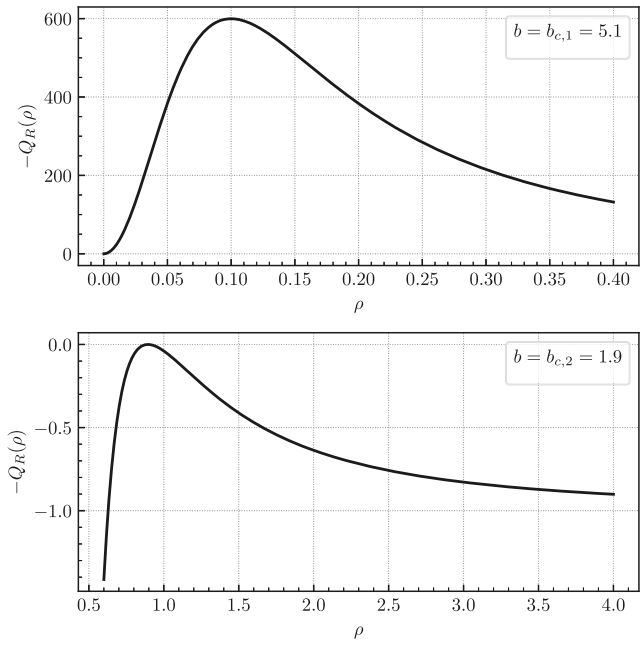

FIG. 4. D1-D5 circular fuzzball radial effective potential in the equatorial plane $\theta = \pi/2$ for $L_1 = L_5 = 1$ and $a_f = 0.1$. Top panel: at $b_c = b_{c,1} = 5.1$, the stable photon sphere corresponds to $\rho_c = \rho_{c,1} = 0$. Bottom panel: at $b_c = b_{c,2} = 1.9$, the unstable photon sphere corresponds to $\rho_c = \rho_{c,2} = \sqrt{0.8} = 0.8944$.

Details can be found in Appendix B. In the first equation of (3.17), the negative sign refers to the phase in which the particle is moving closer to the central object, while the positive sign refers to the opposite situation. We can find the critical geodesics by imposing the vanishing of Eq. (3.15) and its radial derivative. Introducing the impact parameter b, we have two equations in the variables ρ, b, and we can solve them in terms of the parameters L_1, L_5, and a_f. Keeping in mind that ρ and b are non-negative, we obtain the final results

$$\rho_{c,1} = 0,$$
$$b_{c,1} = \frac{a_f^2(L_1^2 + L_5^2) + L_1^2 L_5^2}{2a_f L_1 L_5}, \tag{3.18}$$

and

$$\rho_{c,2} = \sqrt{L_1 L_5 - a_f(L_1 + L_5)},$$
$$b_{c,2} = L_1 + L_5 - a_f. \tag{3.19}$$

Figure 4 shows the plot of the effective radial potential $-P_\rho^2 = -Q_R(\rho)$ for these two cases by setting $L_1 = L_5 = 1$ and $a_f = 0.1$. As can be seen, $\rho_{c,1}$ describes a stable photon sphere (top panel), while $\rho_{c,2}$ describes an unstable one (bottom panel). By plugging $\theta = \pi/2$ and $\rho = 0$ into Eq. (3.8), we obtain

$$\begin{cases} x_1 = a_f \cos\phi, \\ x_2 = a_f \sin\phi, \\ x_3 = x_4 = 0. \end{cases} \tag{3.20}$$

Hence, when $\rho = 0$, we get on the plane (x_1, x_2) a circle of radius a_f, and it is the stable geodesic.[10] In the following sections of the paper, we will indicate (x_1, x_2) as the usual (x, y) and only focus on the orbit with radius $\rho_c = \sqrt{L_1 L_5 - a_f(L_1 + L_5)}$ and an impact parameter equal to $b_c = L_1 + L_5 - a_f$. From the above discussion, it is clear that, when $b < b_c$, the particle, coming from infinity, arrives at the compact object, and then it bounces, while for $b > b_c$, the particle reaches a minimum value of the radius, and then it moves away again. The former case will be denoted as *subcritical*, while the latter is *overcritical*. Another way of formulating this distinction from the first equation of (3.17) is the following:

(1) When $b < b_c$, the radicand does not admit real zeros in ρ. It implies that $\mathrm{d}\rho/\mathrm{d}t$ never vanishes, meaning that the distance always decreases (with minus sign) or always increases (with plus sign). If the particle comes from infinity, then it reaches $\rho = 0$, and then it bounces.
(2) When $b = b_c$, the radicand has two real and positive zeros that are coincident, and they correspond to ρ_c.
(3) When $b > b_c$, the radicand has two distinct real and positive zeros; hence, we have two critical radii, ρ_- and ρ_+, with $\rho_- < \rho_+$. If the particle comes from infinity, it first reaches ρ_+, and after this, it goes away again.

It has to be noticed that, in this case, we can rewrite the equatorial geodesics in terms of elliptic integrals, which is done from Eq. (3.17). This constitutes the ground truth in the comparison of the results of the HNN and the numerical integrators. Details of the determination of the ground truth can be found in Appendix C. In the following sections, in the planar case, we will set $L_1 = L_5$ without loss of generality, simplifying a lot of the involved equations.

E. Nonplanar case

Now, we want to analyze the motion that takes place not only on the equatorial plane. In particular, we set

$$P_z = J_\psi = 0, \tag{3.21}$$

and we fix $\psi = 0$. It corresponds to a family of three-dimensional geodesics that take place in the Cartesian space through the relations outlined in Eq. (3.8)—i.e.,

[10] In the $\theta = 0$ plane described by the coordinates (x_3, x_4) (since $x_1 = x_2 = 0$), the fuzzball instead corresponds to a point located at $x_3 = x_4 = 0$.

$$
\begin{aligned}
x &= \sqrt{\rho^2 + a_f^2}\sin\theta\cos\phi, \\
y &= \sqrt{\rho^2 + a_f^2}\sin\theta\sin\phi, \\
z &= \rho\cos\theta,
\end{aligned}
\tag{3.22}
$$

where we have denoted x_3 as z. The general Hamiltonian (3.9) now reads

$$
\begin{aligned}
\mathcal{H} = \frac{1}{2(\rho^2 + a_f^2\cos^2\theta)H}\Bigg[& P_\rho^2(\rho^2 + a_f^2) + P_\theta^2 \\
& + \frac{J_\phi^2}{\sin^2\theta} - \frac{(J_\phi a_f - E L_1 L_5)^2}{\rho^2 + a_f^2} \\
& - E^2(\rho^2 + a_f^2 + L_1^2 + L_5^2 - a_f^2\sin^2\theta)\Bigg].
\end{aligned}
\tag{3.23}
$$

The derivatives with respect to the variables that define the Hamilton equations are reported in Appendix A.

We discuss now the determination of the critical geodesics—i.e., the only case we study with the HNN, since it is the most physically relevant one for observational reasons. The momenta (3.10) and (3.11) now become

$$
P_\rho^2 = E^2\left(1 + \frac{L_1^2 + L_5^2}{\rho^2 + a_f^2}\right) + \frac{(J_\phi a_f - E L_1 L_5)^2}{(\rho^2 + a_f^2)^2} - \frac{K^2}{\rho^2 + a_f^2}
\tag{3.24}
$$

and

$$
P_\theta^2 = K^2 - \frac{J_\phi^2}{\sin^2\theta} - E^2 a_f^2\sin^2\theta,
\tag{3.25}
$$

respectively. For $P_\theta \propto \dot{\theta} \neq 0$, the angular dynamics is nontrivial. Let us rewrite (3.24) as follows:

$$
\begin{aligned}
\frac{(\rho^2 + a_f^2)^2 P_\rho^2}{E^2} &= \mathcal{R}(\rho) = \rho^4 + A\rho^2 + B, \\
A &= 2a_f^2 + L_1^2 + L_5^2 - b^2, \\
B &= a_f^2(\zeta^2 - b^2 + a_f^2 + L_1^2 + L_5^2),
\end{aligned}
\tag{3.26}
$$

where

$$
\zeta = b_\phi - \frac{L_1 L_5}{a_f}, \qquad b_\phi = \frac{J_\phi}{E}, \qquad b = \frac{K}{E}.
\tag{3.27}
$$

The critical conditions are $\mathcal{R}(\rho_c) = \mathcal{R}'(\rho_c) = 0$, and these correspond to two equations in terms of the variables ρ_c, ζ_c, and b_c. We can solve them in terms of ζ_c and b_c, which become functions of ρ_c—i.e.,

$$
\begin{aligned}
\zeta_c^2 &= \frac{(\rho_c^2 + a_f^2)^2}{a_f^2}, \\
b_c^2 &= 2\rho_c^2 + 2a_f^2 + L_1^2 + L_5^2.
\end{aligned}
\tag{3.28}
$$

In order to fix ρ_c, we must work on the condition $P_\theta^2 \geq 0$. Details of all steps that have to be done in order to arrive at the determination of ρ_c and the examination of the corresponding critical geodesics are reported in Appendix D. Indeed, as shown there, these critical geodesics generically wrap around spheroidal zones. There is only one case in which the critical geodesics wrap around an oblate spheroid, and we will study only this last scenario. The critical radius has the following expression:

$$
\rho_c = \sqrt{L_1 L_5 - a_f^2}, \qquad 0 < a_f \leq \sqrt{L_1 L_5}.
\tag{3.29}
$$

These geodesics are *counterrotating* with respect to the fuzzball—i.e., they rotate in the opposite sense with respect to the central object. From Eq. (3.29), we can obtain the expressions of ζ_c and b_c, which are

$$
\begin{aligned}
\zeta_c &= -\frac{L_1 L_5}{a_f} \leftrightarrow b_{\phi,c} = 0, \\
b_c &= L_1 + L_5.
\end{aligned}
\tag{3.30}
$$

The reason why we have the negative sign of Eq. (3.28) for ζ_c is explained in Appendix D.

IV. RESULTS FOR PLANAR GEODESICS

In this section, we present in detail the implementation of the HNNs, the strategies, and the numerical results for the planar geodesics in the $\theta = \pi/2$ plane and in the three scenarios illustrated in Sec. III D: overcritical ($b > b_c$), critical ($b = b_c$), and subcritical ($b < b_c$). The Hamilton equations that describe the geodesics are given by the following system of coupled differential equations:

$$
\begin{aligned}
\dot{\rho} &= \frac{\partial\mathcal{H}}{\partial P_\rho}, \\
\dot{P}_\rho &= -\frac{\partial\mathcal{H}}{\partial\rho}, \\
\dot{\phi} &= \frac{\partial\mathcal{H}}{\partial J_\phi}.
\end{aligned}
\tag{4.1}
$$

The derivatives on the right-hand side are explicitly written in Appendix A. The integration of the equations obtained from the separability of the motion [Eq. (3.17)], which are different but physically equivalent, will be presented in Appendix E.

For all three cases, we fix the geometry parameters to $a_f = 0.1$, $L_1 = L_5 = L = 1$, and $E = 1$. With this choice,

the critical values of the impact parameter (3.19) and of the critical radius are

$$b_c = 1.9, \qquad \rho_c = \sqrt{0.8} = 0.8944. \tag{4.2}$$

In the remainder of this section, we outline the general strategy, while the case-dependent details and the associated results will be presented separately in the following three subsections.

The neural networks are explicitly designed to map the input time s to the three-dimensional output $[O_\rho(s), O_{P_\rho}(s), O_\phi(s)]$, from which the solutions are built according to

$$\begin{aligned} \hat{\rho}(s) &= \rho_0 + f(s) O_\rho(s), \\ \widehat{P_\rho}(s) &= P_{\rho,0} + f(s) O_{P_\rho}(s), \\ \hat{\phi}(s) &= \phi_0 + f(s) O_\phi(s). \end{aligned} \tag{4.3}$$

Let us stress that the role of the time t in Sec. II A is now played by the affine parameter s, and this does not entail any difference, either in the strategy or in the implementation of the neural networks. Indeed, Sec. II A can be reread replacing t with s.

Our implementation of the neural networks[11] relies on the libraries Keras [56] and TensorFlow [57]. The trainings are performed by employing the Adam optimizer of Ref. [58]. In all cases, the weights are initialized by drawing random numbers from a normal distribution with center 0 and width 0.05.

The loss function we minimize is the combination given in Eq. (2.15), and during the training we monitor individually also L_{dyn} and L_{energy}, which are defined in Eqs. (2.11) and (2.14), respectively. Unless otherwise specified, we set $\gamma_\rho = \gamma_{P_\rho} = \gamma_\phi = 1$ inside the loss function (2.11). As usually happens when training a neural network, the loss function fluctuates continuously over the epochs. Therefore, we save the updated weights at the end of each epoch and, at the end of the training, we consider the set of weights corresponding to the lowest value reached by the loss function as the optimal set for the trajectory prediction.

If not otherwise specified, the learning rate η is defined as a function of the number of the epoch according to the scheduler function:

$$\eta(\text{epoch}) = \eta_f + \frac{\eta_i - \eta_f}{1 + \exp[-(\eta_c - \text{epoch})/\sigma_\eta]}, \tag{4.4}$$

with $\eta_i = 8 \times 10^{-4}$, $\eta_f = \eta_i/10$, $\eta_c = 10^5$, and $\sigma_\eta = 20 \times 10^3$. It can be easily seen that Eq. (4.4) is a decreasing function such that $\eta \sim \eta_i$ for epoch $\ll \eta_c$ and $\eta \sim \eta_f$ for epoch $\to \infty$. Setting a decreasing learning rate during the training is a standard technique to improve the convergence toward the minimum of the loss function, and the choice of the aforementioned parameters has been made after a careful investigation.

In addition, for each case we perform several trainings by changing λ and the neural network architecture, and we judge the best choice according to the lowest loss function. We will show an example of this study for the overcritical case, and for the other cases (including the nonplanar motion in Sec. V), we will only quote the configuration that according to our analysis is the one that leads to the best performance.

As explained in Sec. III D from Eq. (3.8), the solution in the x-y Cartesian plane (recall that now $\theta = \pi/2$) is given by

$$\begin{aligned} x &= \sqrt{\rho^2 + a_f^2}\cos\phi, \\ y &= \sqrt{\rho^2 + a_f^2}\sin\phi. \end{aligned} \tag{4.5}$$

For the three cases, we set the initial value of the radius ρ_0 to 10. The initial values of the variables P_ρ and ϕ depend instead on the choice of the value of the impact parameter b. On the Cartesian plane, the impact parameter b is identified with y_0, and therefore the initial value of the angular variable is given by inverting the second equation of (4.5)—i.e.,

$$\phi_0 = \arcsin\left(\frac{b}{\sqrt{\rho_0^2 + a_f^2}}\right). \tag{4.6}$$

The value of $P_{\rho,0}$ can be obtained by solving $\mathcal{H}(0) = 0$ in Eq. (3.13).

In each case, we set the interval of discrete times in correspondence of which the loss function (2.15) has to be minimized by splitting the interval $\mathcal{T}_{\text{train}} = [0, T]$ into N equally spaced points. We keep this set fixed during the entire training process. We then construct a second interval of discrete times, denoted by $\mathcal{T}_{\text{val}}$, by adding five additional equally spaced points between two consecutive times in $\mathcal{T}_{\text{train}}$ and excluding the times in $\mathcal{T}_{\text{train}}$. The interval $\mathcal{T}_{\text{val}}$ thus contains $5 \times (N-1)$ points which are not used during the training. For all the trainings performed in this work, we have explicitly checked that, at the end of the training, the loss function evaluated for the set $\mathcal{T}_{\text{val}}$ is compatible with that actually minimized within 5%. This is a numerical check that overfitting of the solution has not occurred. Finally, we show in the plots the results predicted for the joint interval $\mathcal{T}_{\text{train}} \cup \mathcal{T}_{\text{val}}$, which contains $N + 5 \times (N-1)$ points in total.

In order to assess the performance of the neural networks, we compare our results with those obtained by other

[11] A demonstrative notebook is provided in Ref. [55]. The complete code is available upon request.

numerical integrators. In particular, we consider the first-order semi-implicit Euler integrator (2.6) that, as is well known, preserves the symplectic structure of Hamiltonian systems by conserving the energy up to a shift term proportional to the discretization step, and the more advanced Runge-Kutta method of order 5(4), `RK45` in short, though it is not symplectic. To implement the latter, we use the SciPy package `solve_ivp`. The package (see the documentation in Ref. [59]) conveniently allows us to set the precision of the solution $y(t)$ in terms of absolute and relative tolerances, determined by the parameters $a_{\rm tol}$ and $r_{\rm tol}$, respectively, by keeping the local error estimates smaller than $a_{\rm tol} + r_{\rm tol}|y(t)|$. We have set in all cases the relative and absolute tolerances to the minimum possible values, $r_{\rm tol} = 10^{-15}$ and $a_{\rm tol} = 10^{-15}$, respectively.

We evaluate the numerical solution obtained from `RK45` in correspondence with the interval $\mathcal{T}_{\rm train} \cup \mathcal{T}_{\rm val}$ as well. The numerical solution provided by the Euler method is instead strongly dependent on the discretization step Δs, which regulates the update of the solution in Eq. (2.6). We consider two time steps: $\Delta s = T/N$, which corresponds to the same step as $\mathcal{T}_{\rm train}$, and $\Delta s = T/(100N)$, which is instead 100 times finer. In the discussion of the results, we will refer to these different Euler solutions with $1 \times N$ and $100 \times N$ points, respectively.

As explained in Appendix C, we can compute ϕ exactly as a function of ρ, and then, by plugging these values into Eq. (4.5), we get the exact trajectory in the Cartesian plane. We call this trajectory the "ground truth," as is customary in machine learning studies. The comparison with the ground truth only allows for a qualitative check of the correctness of the predicted geodesics. In order to have a quantitative measure of the error committed by the neural network and the numerical integrators, we can compare the exact function $\phi(\hat{\rho})$ with $\hat{\phi}(\hat{\rho})$, and $\phi(\tilde{\rho})$ with $\tilde{\phi}(\tilde{\rho})$. The same comparison can be carried out also for P_ρ, whose true value can be found as function of ρ by solving $\mathcal{H}(s) = 0$, as is explicitly done in Eqs. (C5)–(C7).

We pass now to examine separately the three regimes.

A. Overcritical case

In this subsection, we discuss the results for the overcritical case and set $b = 1.91$. The initial conditions are therefore given by

$$\begin{aligned} \phi_0 &= 0.1922, \\ \mathcal{H}(0) &= 0, \\ P_{\rho,0} &= -0.9917. \end{aligned} \tag{4.7}$$

To find the optimal setup, we have repeated the training several times by varying the trade-off parameter λ and the number of neurons $N_{\rm neurons}$ while keeping fixed the number of hidden layers at 2 and the activation function $\sigma(x)$ as the hyperbolic tangent,

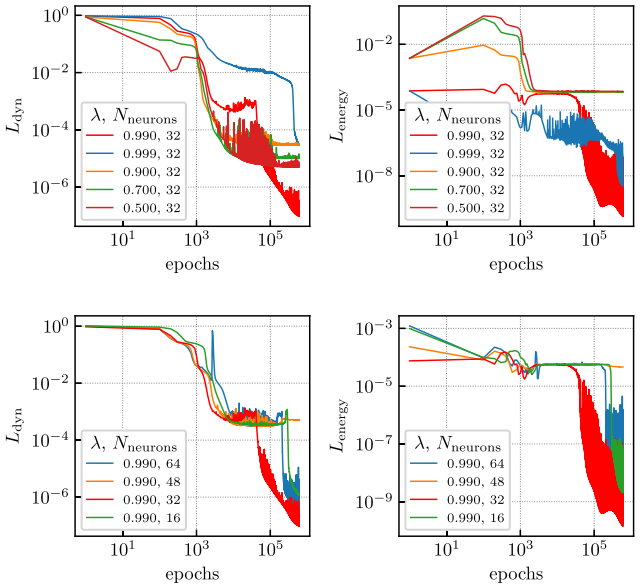

FIG. 5. Tuning of the parameters for the overcritical case with $b = 1.91$. In all the tests, the neural network has two hidden layers, and $\tanh(x)$ is the activation function. Top panels: $L_{\rm dyn}$ (left) and $L_{\rm energy}$ (right) at a fixed number of 32 neurons while changing λ from 0.5 to 0.99. Bottom panels: the same as the top panel, but with fixed $\lambda = 0.99$ while varying the number of neurons from 16 to 64. The four plots show that the best performance, in terms of minimization of the loss function, is given by the choice $\lambda = 0.99$ and $N_{\rm neurons} = 32$, corresponding to the red color.

$$\sigma(x) = \tanh(x). \tag{4.8}$$

For λ, we have considered the values 0.5, 0.7, 0.9, 0.99, and 0.999, while for $N_{\rm neurons}$, we have considered the values 16, 32, 48, and 64. The interval of time $\mathcal{T}_{\rm train}$ is given by splitting the interval [0, 24] into $N = 400$ points. The neural network is trained with 600×10^3 epochs for each combination of parameters. The loss functions $L_{\rm dyn}$ and $L_{\rm energy}$ are displayed in Fig. 5, where the top panel shows the dependence on λ at fixed $N_{\rm neurons} = 32$, while the bottom panel shows the dependence on $N_{\rm neurons}$ at fixed $\lambda = 0.99$. The study shows a strong dependence upon both λ and $N_{\rm neurons}$. In particular, at fixed $N_{\rm neurons} = 32$, the loss functions get stuck in a plateau for $\lambda \leq 0.9$, highlighting that indeed, the addition of the conservation term $L_{\rm energy}$ favors a faster and better minimization of the total loss function. We have observed no significant improvement by passing from two to three hidden layers.

The study shows that the best performance is achieved by the combination $\lambda = 0.99$ and $N_{\rm neurons} = 32$. We have then trained the neural network with this combination up to 3×10^6 epochs, and we have used it to predict the geodesic. The corresponding loss functions are shown in Fig. 6.

Figure 7 shows the function $\mathcal{H}(s)$ for the predicted geodesic (labeled by "HNN" and represented by the solid

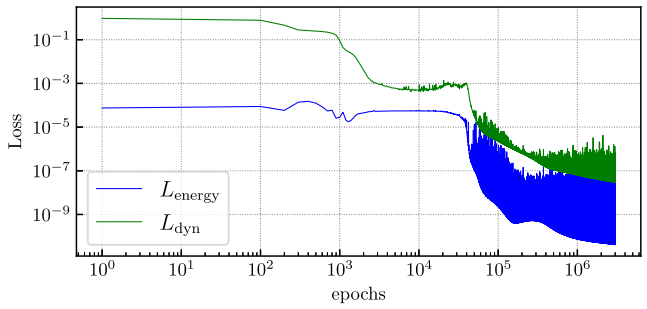

FIG. 6. Loss functions for the overcritical case with $b = 1.91$. The training has been performed with $\lambda = 0.99$, $N_{\text{neurons}} = 32$, two hidden layers, and $\tanh(x)$ as the activation function.

red line), for the Runge-Kutta method (purple dashed line), and for the Euler method with N (blue) and $100 \times N$ (green) points. The plot shows that the HNN conserves the energy very well on average, with fluctuations between -2×10^{-5} and 2×10^{-5}. The Euler methods both with N and with $100 \times N$ points exhibit a larger violation of the energy that increases with the time s. The `RK45` integrator is remarkably good, with violations from $\mathcal{H}(0) = 0$ of order 10^{-9}.

Before showing the results for $\rho(s)$, $P_\rho(s)$, and $\phi(s)$, let us discuss their expectations based on the physical arguments of Sec. III D. At the beginning of the motion, the particle approaches the fuzzball (we refer to this phase as "incoming motion"), and consequently ρ decreases until it reaches the minimum value $\rho_+ = 1.000138$, which is the largest root of the radicand in Eq. (3.17). Afterward, the particle moves away from the fuzzball (we refer to this phase as "outgoing motion"), and ρ starts increasing again. We denote by s_+ the time at which the variable ρ attains its minimum ρ_+. Due to the symmetry of the system under $\rho \mapsto -\rho$, the function $\rho(s)$ must be symmetric with respect to s_+. On the other hand, P_ρ starts from the negative value $P_{\rho,0}$, given in Eq. (4.7), and it increases over time until it reaches zero at s_+, eventually approaching a plateau at infinite time. Similarly, ϕ starts from the initial value ϕ_0

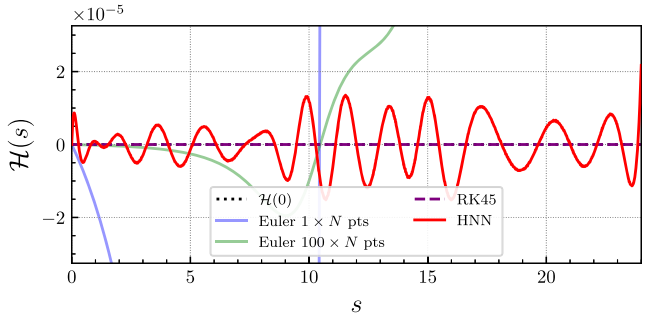

FIG. 7. Energy conservation over time in the overcritical case with $b = 1.91$ and $N = 400$ for different methods. The conserved value of the energy $\mathcal{H}(0) = 0$ is represented by the dotted black line.

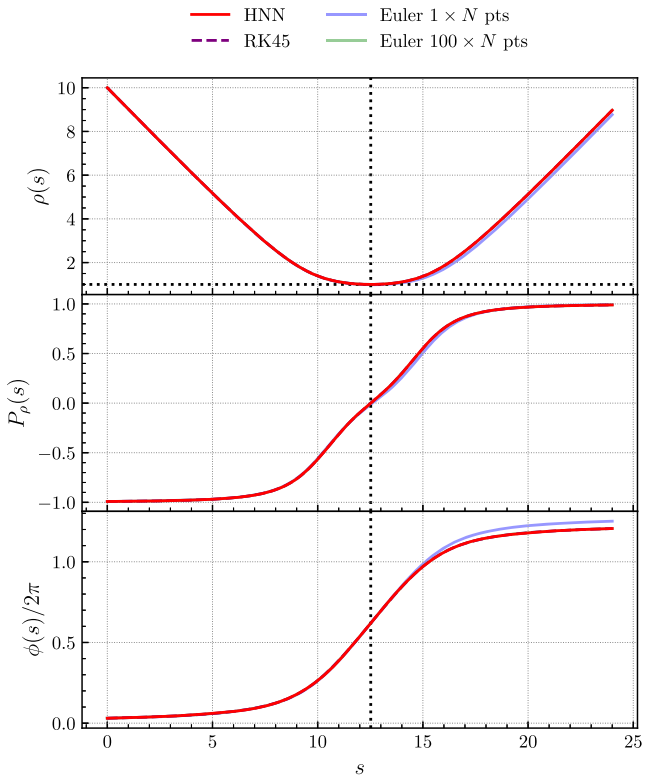

FIG. 8. From top to bottom: $\rho(s)$, $P_\rho(s)$, and $\phi(s)/2\pi$ for different methods in the overcritical case with $b = 1.91$ and $N = 400$. The vertical dotted line in the three panels represents $\hat{s}_+ = 12.521$—i.e., the time at which the minimum of ρ is reached, and with respect to which the motion is symmetric. The motions to the left and to the right of $\hat{s}_+$ are incoming and outgoing, respectively. The horizontal dotted line in the top panel represents $\rho_+ = 1.000138$.

written in Eq. (4.7), then increases and asymptotically reaches a plateau at infinite time.

This qualitative description of the motion is numerically confirmed by the functions $\rho(s)$, $P_\rho(s)$, and $\phi(s)/2\pi$ that are shown in Fig. 8 for different methods. The three dynamical variables behave qualitatively as expected, and there is no qualitative difference between the predictions of the HNN, `RK45`, and the Euler method with $100 \times N$ points, while Euler with N points visibly deviates from the others. From the numerical solution of the HNN, we find that the minimum value of ρ is $\hat{\rho}_+ = 1.000153$, 0.002% away from ρ_+, at time $\hat{s}_+ = 12.521$.

In order to have a quantitative measure of the quality of the solutions predicted by the HNN, we consider the quantities

$$\delta P_\rho(\hat{\rho}) = \widehat{P_\rho}(\hat{\rho}) - P_\rho(\hat{\rho}) \tag{4.9}$$

and

$$\delta\phi(\hat{\rho}) = \hat{\phi}(\hat{\rho}) - \phi(\hat{\rho}). \tag{4.10}$$

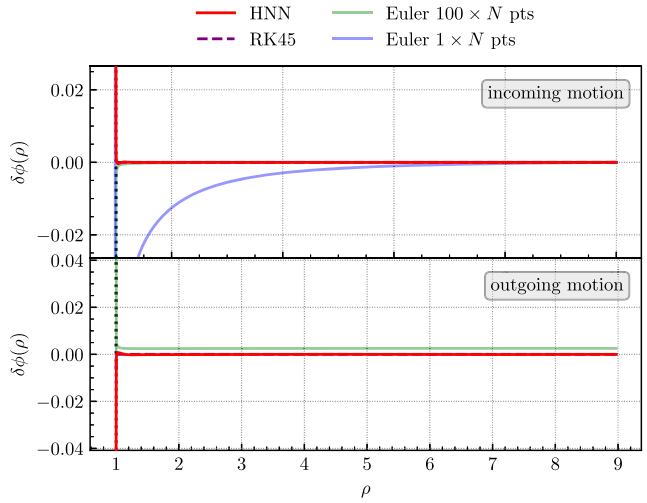

FIG. 9. Error on the function $\phi(\rho)$ for different methods, and for the incoming motion (top panel) and the outgoing motion (bottom panel) in the overcritical case with $b = 1.91$. The vertical dotted line represents $\rho_+ = 1.000138$.

Equivalent quantities can be considered also for the solutions obtained from numerical integrators. As anticipated before, the functions $P_\rho(\rho)$ and $\phi(\rho)$ can be computed exactly, and therefore Eqs. (4.9) and (4.10) are good indicators to quantify the error made by a given method. For this reason, we will refer to these quantities as "errors" henceforth. The functions $\phi(\rho)$ and $P_\rho(\rho)$ are obtained as explained in Appendix C.

The errors of the HNN and the numerical integrators are shown in Figs. 9 and 10 for the variables ϕ and P_ρ, respectively. Both the figures show the error in the incoming motion (top panel) and outgoing motion (bottom panel). As can be seen, the error of the HNN on the variable ϕ oscillates uniformly around 0 with an order of magnitude of 10^{-4}, and with a spike occurring for ρ close to ρ_+. An analogous picture is found for the variable P_ρ, but with an order of magnitude of 10^{-5}. The small scale of the errors [compare with $P_\rho(\rho)$ and $\phi(\rho)$ in Figs. 35 and 36 in Appendix C] is a reassuring evidence that the Hamilton equations have been solved by the HNN with a high accuracy. The Euler method, both with N and with $100 \times N$ points, has larger errors than HNN. `RK45` is instead the most precise, with errors that fluctuate on the scales 10^{-9} and 10^{-10} for ϕ and P_ρ, respectively.

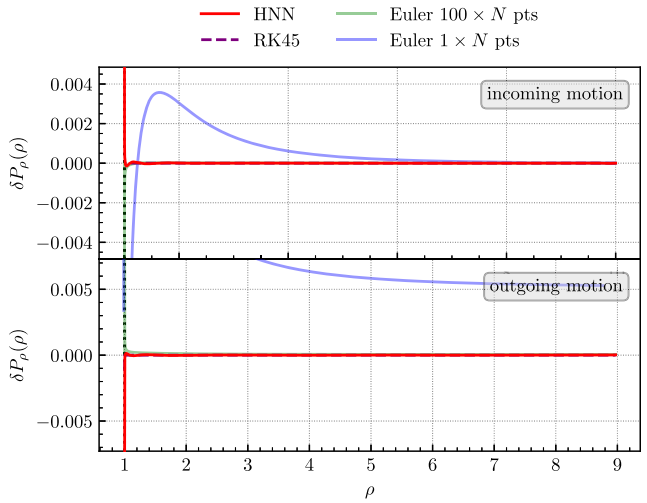

FIG. 10. Error on the variable $P_\rho(\rho)$ for different methods and for the incoming motion (top panel) and the outgoing motion (bottom panel) in the overcritical case with $b = 1.91$. The vertical dotted line represents $\rho_+ = 1.000138$.

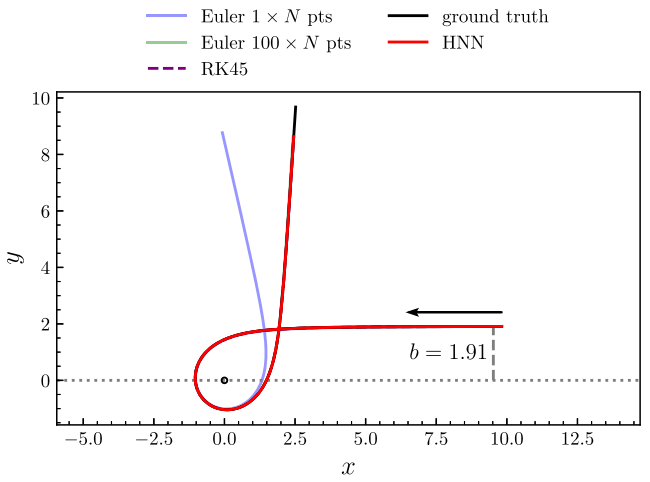

FIG. 11. Geodesics in the Cartesian plane obtained from different methods in the overcritical case with $b = 1.91$ and $N = 400$. The ground truth is represented by the solid black line, while the small circle at the center (0,0) has radius a_f and represents the fuzzball. The black arrow indicates the incoming direction of the particle.

We eventually project the motion obtained for these different methods to the Cartesian plane by using Eq. (4.5), and we compare it to the ground truth. The geodesics in the Cartesian plane are shown in Fig. 11. The HNN and `RK45` have no visible differences with respect to the ground truth, showing that they are both efficient integrators of the Hamilton equation of motion. Concerning the Euler method, for the same number of points as those used to train the neural network, the trajectory shows a clear deviation from the ground truth over long timescales, and a reliable result is obtained only by considering a number of points (time step) larger (smaller) by a factor of 100.

We conclude this subsection by employing the HNN to compute the geodesics for impact parameters $b = 1.95$, 2.0, and 2.2. These three cases have been obtained by considering the same setup as $b = 1.91$ discussed above—i.e., $\lambda = 0.99$, $N_{\text{neurons}} = 32$, $N = 400$, $\mathcal{T}_{\text{train}} = [0, 24]$, and 3×10^6 epochs. For all three cases, we have computed the errors and checked that the quality of the solution is at the

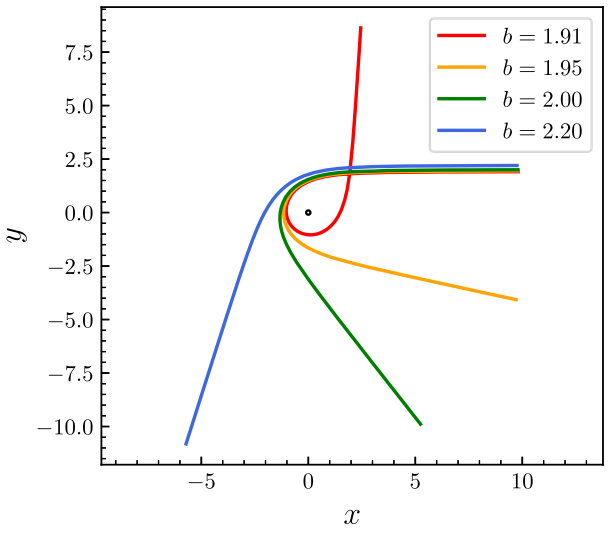

FIG. 12. Geodesics in the Cartesian plane predicted by the HNN for different values of the impact parameter b.

same level as the case with $b = 1.91$. In Fig. 12, we show the final geodesics predicted by the HNN in the Cartesian plane. As can be seen from the plot, the scattering angle at which the particle moves away from the fuzzball increases as the impact parameter increases, and this is expected by theory (in fact, as the impact parameter increases, the particle is less affected by the gravitational field of the fuzzball, and therefore its trajectory is less bent). We leave to a future work the intriguing possibility to treat the impact parameter b not as a fixed algorithmic parameter but as an additional variable to feed to the neural network in input together with the time.

B. Critical case

In this subsection, we present the critical case obtained by setting $b = b_c = 1.9$. The initial conditions now read

$$\begin{aligned} \phi_0 &= 0.1912, \\ \mathcal{H}(0) &= 0, \\ P_{\rho,0} &= -0.9919. \end{aligned} \tag{4.11}$$

For this case, we have considered the time interval $\mathcal{T}_{\text{train}} = [0, 80]$ sampled with $N = 800$ points. The reason behind such a long time interval is to probe the stability of the trajectory on the critical radius circumference over long timescales. We have repeated the tuning procedure discussed in the overcritical case and found that the optimal setup is given by $\lambda = 0.99$ and $N_{\text{neurons}} = 48$. The number of hidden layers is two, and the activation function is given by Eq. (4.8). The neural network is trained with 3×10^6 epochs, and the corresponding loss functions L_{dyn} and L_{energy} are shown in Fig. 13.

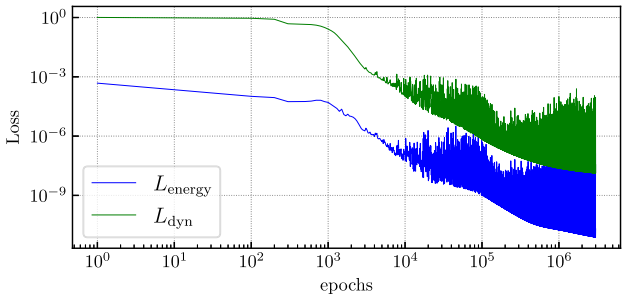

FIG. 13. Loss functions for the critical case with $b = b_c = 1.9$. The training has been performed with $\lambda = 0.99$, $N_{\text{neurons}} = 48$, two hidden layers, and $\tanh(x)$ as the activation function.

In Fig. 14, we show the function $\mathcal{H}(s)$ for the different methods. The energy is very well conserved on average by the HNN, with oscillations of order 10^{-5} for s smaller than $\simeq 14$ and of order 10^{-9} for larger times. The Euler method with N and with $100 \times N$ points does not conserve the energy from the very beginning. Particular attention should be paid to `RK45`, which conserves the energy on a scale of 10^{-13}. This information seems to indicate the `RK45` method as the best for trajectory prediction. However, as we will see shortly, exact energy conservation by a nonsymplectic method does not automatically imply an exact solution of the equations of motion.

Concerning the physics of the motion, in this case we only have an incoming phase in which the radial coordinate $\rho(s)$ decreases from ρ_0 to reach asymptotically the value $\rho_c = 0.8944$. Correspondingly, the radial momentum $P_\rho(s)$ increases from the negative value $P_{\rho,0}$ given in Eq. (4.11) up to freezing at 0. The function $\phi(s)$ is instead expected to show linear growth as the particle stabilizes, rotating on the critical geodesic.

Figure 15 shows the functions $\rho(s)$, $P_\rho(s)$, and $\phi(s)/2\pi$ for HNN and the numerical integrators. As can be seen, the HNN is the only method that correctly reproduces the expected dynamics of the motion. The Euler method with both N and $100 \times N$ points starts failing around $s \simeq 15$ and exhibits badly oscillating behavior. The `RK45` method,

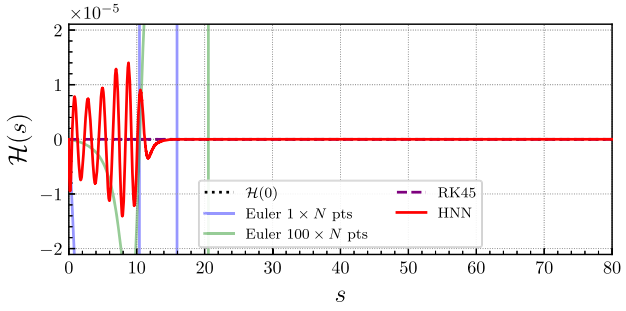

FIG. 14. Energy conservation over time in the critical case with $b = b_c = 1.9$ and $N = 800$ for different methods. The conserved value of the energy $\mathcal{H}(0) = 0$ is represented by the dotted black line.

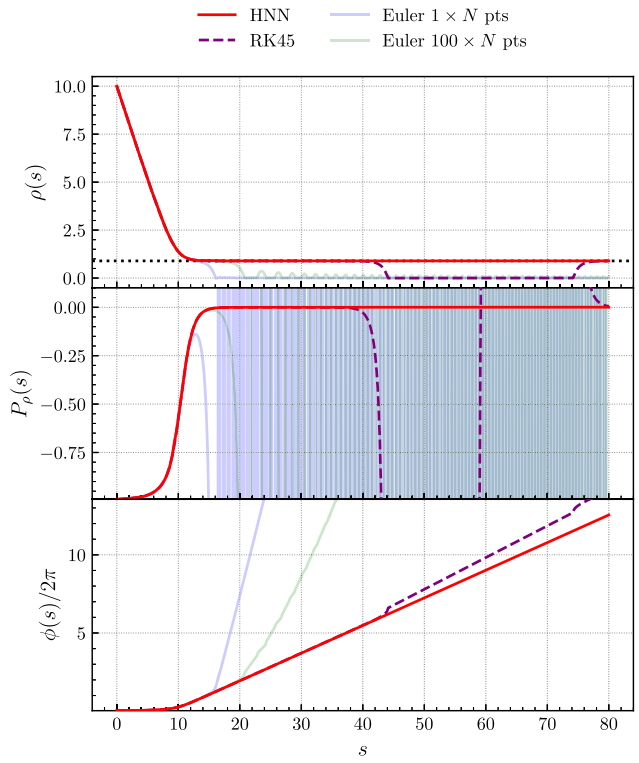

FIG. 15. From top to bottom: $\rho(s)$, $P_\rho(s)$, and $\phi(s)/2\pi$ for different methods in the critical case with $b = b_c = 1.9$ and $N = 800$. The horizontal black dotted line in the top panel represents the critical radius $\rho_c = 0.8944$, on which the trajectory must settle. The trajectory predicted by the HNN is the only one among the different methods to correctly reproduce this feature.

despite exactly conserving energy, fails for $s > 44$. In particular, $\rho(s)$ for both Euler and `RK45` approaches 0, which corresponds to a situation in which the particle trajectory has collapsed onto the sphere of radius a_f, and it is physically impossible for this impact parameter. The variable $\phi(s)$ predicted by the HNN exhibits the correct linear growth for times larger than $s \simeq 22$, which we can consider as the time at which the trajectory stabilizes on the critical circumference. The angle $\phi(s)$ is plotted normalized by 2π in order to give an idea of the number of windings of the trajectory around the fuzzball. From time $s = 22$ to time $s = 80$, the trajectory completes approximately ten full windings and remains stable throughout.

Since numerical integrators clearly fail, it does not make sense to compare the error of the HNN with the error of the integrators. Rather, to assess the stability of the solution predicted by the neural network, we show in Fig. 16 the quantities $\hat{\rho}(s) - \rho_c$ and $\widehat{P_\rho}(s)$ for large times. Both quantities oscillate around 0 on a remarkably small scale. A small deviation from 0 is observed for $s \simeq 76$. We will investigate the cause of this effect in the future, but it is possible that it could be reduced simply by increasing the number of epochs. As can be seen in Fig. 13, in fact, the loss functions have not yet reached their minimum.

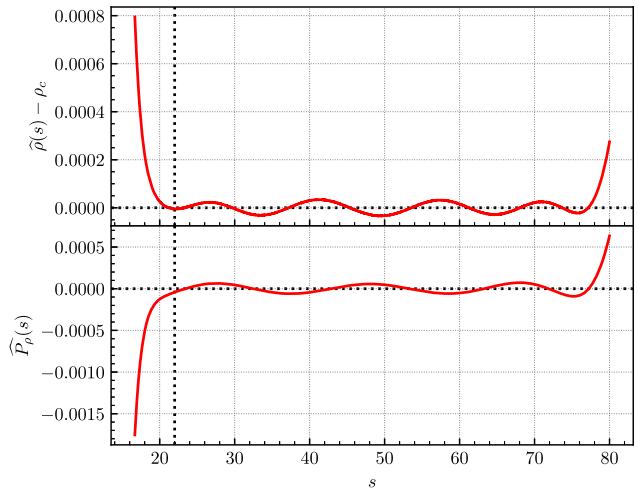

FIG. 16. Stability of the trajectory predicted by the HNN over a long timescale. The vertical dotted line is drawn at $s \simeq 22$, the time at which the trajectory approximately reaches the critical circumference. Top panel: difference between $\hat{\rho}(s)$ and $\rho_c = 0.8944$. Bottom panel: predicted $\widehat{P_\rho}(s)$.

At this point, we emphasize that although the `RK45` method is very accurate and the Euler method is symplectic, the HNN is the only approach that correctly integrates the equations of motion, outperforming both methods in terms of efficiency and accuracy. The HNN's ability to predict a stable trajectory over long timescales is due to the introduction of the energy conservation term, which penalizes solutions that violate this constraint. The ability to enforce constraints within the machine learning framework proves to be an extremely beneficial factor in this case, enabling accurate results where standard methods fail.

We then project the motion to the Cartesian plane by means of Eq. (4.5), and the result is shown in Fig. 17, together with the ground truth. As can be appreciated, the geodesic predicted by the HNN remains stable along the circumference of radius $\sqrt{\rho_c^2 + a_f^2}$, while all the other methods predict a geodesic that collapses onto the fuzzball.

C. Subcritical case

In this subsection, we present the results for the last regime—i.e., the subcritical one—and we study the situation with $b = 1.89$. The initial conditions are

$$\begin{aligned} \phi_0 &= 0.1901, \\ \mathcal{H}(0) &= 0, \\ P_{\rho,0} &= -0.9921. \end{aligned} \tag{4.12}$$

This case is particularly challenging from the HNN perspective, as the underlying physics involves an impact of the particle against the fuzzball; hence, the radius ρ decreases from ρ_0 down to zero [this can be also deduced

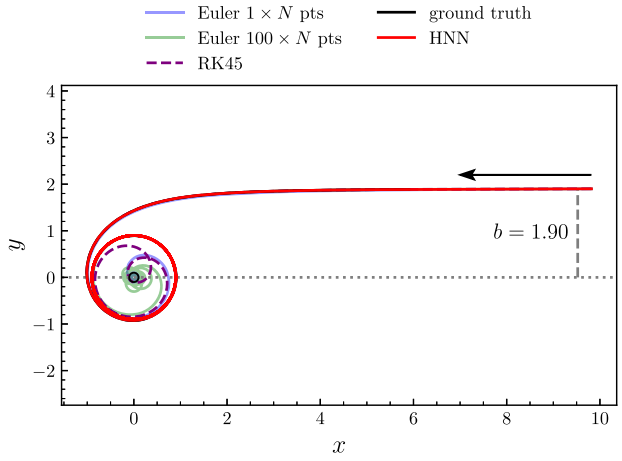

FIG. 17. Geodesics in the Cartesian plane obtained from different methods in the critical case with $b = b_c = 1.9$ and $N = 800$. The ground truth is represented by the solid black line, while the small circle at the center (0,0) has radius a_f and represents the fuzzball. The black arrow indicates the incoming direction of the particle.

from the first equation of (3.17), since the radicand never vanishes]. As a result, the conjugate momentum P_ρ increases in absolute value until it reaches the value

$$\overline{P_\rho} \equiv P_\rho(\rho = 0) = -80.1249, \tag{4.13}$$

where we have used $P_\rho(\rho)$ in Eq. (C7). The problem arises from the fact that the decrease from $P_{\rho,0}$ to $\overline{P_\rho}$ is very rapid in a small interval of values of ρ that are close to 0, and this is shown in the right panel of Fig. 36. The same behavior is observed in the time domain, as can be seen beforehand in the central panel of Fig. 21. From this perspective, the differential equation governing the dynamics of P_ρ is a stiff equation and, as is well known, stiff equations can be a challenge to numerical methods, HNN included. Indeed, we did not manage to obtain accurate solutions by using the same strategy adopted in the past two subsections. Therefore, in the following, we propose two alternative approaches to overcome this difficulty and get a reliable prediction for the geodesic.

1. Approach 1: Splitting the time interval

In the first approach, we split the time interval over which we want to compute the solution in such a way as to separate the stiff regime of P_ρ from one in which its decrease is smoother. This can be easily achieved by dividing the total time interval into consecutive subintervals, and we will refer to this as the "time-splitting approach". For each subinterval, a neural network is trained from scratch using as initial conditions the values of the predicted variables at the final time of the previous interval. We consider the following three time intervals: $\mathcal{T}^{(1)}_{\text{train}} = [0, 14]$, for which the initial conditions are given in Eq. (4.12); $\mathcal{T}^{(2)}_{\text{train}} = [14, 14.8]$, for which the initial conditions are $\rho_0 = 0.6552$, $P_{\rho,0} = -0.8982$, and $\phi_0 = 5.8452$; and finally $\mathcal{T}^{(3)}_{\text{train}} = [14.8, 15.22]$, for which the initial conditions are $\rho_0 = 0.3014$, $P_{\rho,0} = -7.0688$, and $\phi_0 = 7.2790$. All three intervals are divided into 800 points for a total number of $N = 2400$ points. Notice that, at a practical level, the time interval to feed to the neural network always starts at time 0—i.e., $\mathcal{T}^{(2)}_{\text{train}}$ and $\mathcal{T}^{(3)}_{\text{train}}$ actually are [0, 0.8] and [0, 0.42]. Setting as initial conditions the values predicted at the end of the previous interval is what guarantees that the predictions on the subintervals glued together eventually correspond to the solution in the overall interval: $\mathcal{T}_{\text{train}} = [0, 15.22]$.

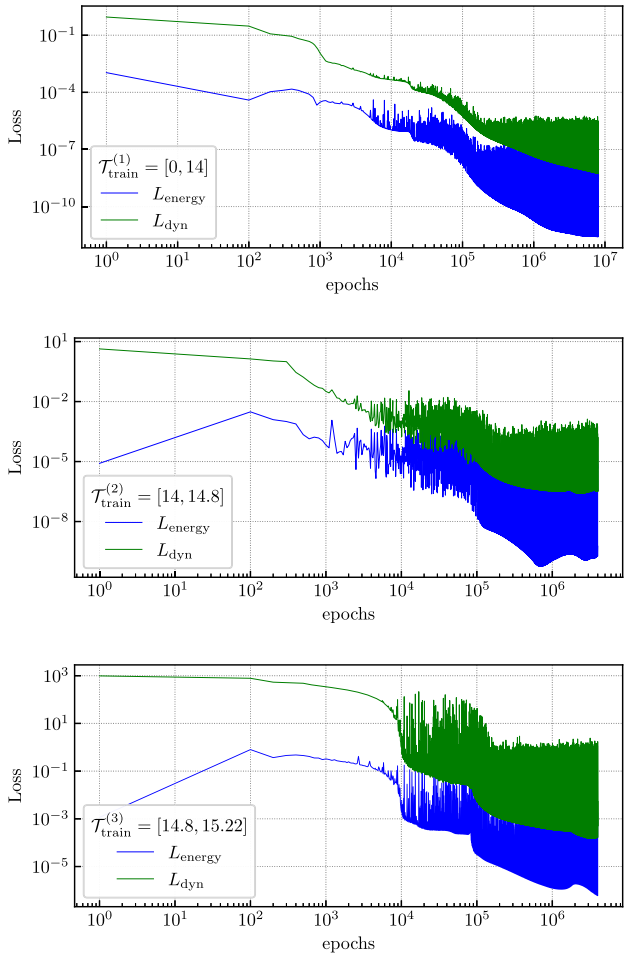

FIG. 18. Loss functions for the subcritical case with $b = 1.89$. From top to bottom, the losses refer to the training intervals $\mathcal{T}_{\text{train}} = [0, 14]$, [14, 14.8], and [14.8, 15.22].

Figure 18 shows the loss functions for the three time intervals. The numbers of epochs, respectively, are 8×10^6, 4×10^6, and 4×10^6. The neural networks in all three cases consist of three hidden layers with 64 neurons each. $\lambda = 0.99$ again, and the activation function is always $\tanh(x)$.

The final time $s = 15.22$ has been chosen such that ρ reaches and slightly exceeds the value 0. We call $\bar{s}$ the time

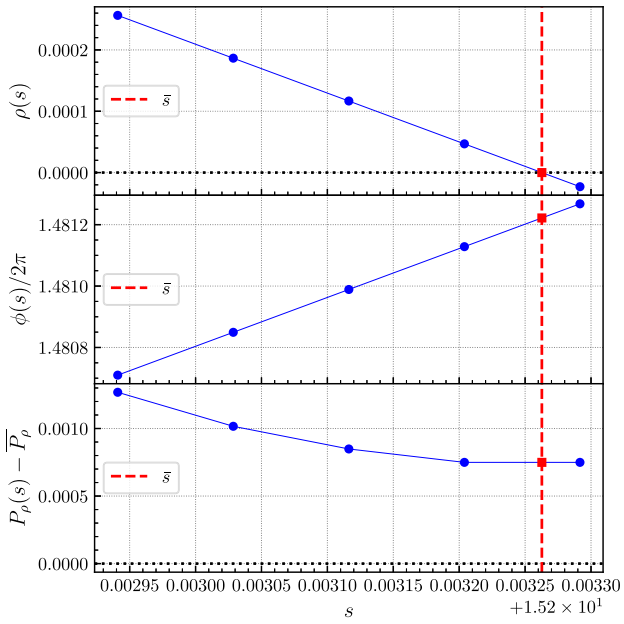

FIG. 19. From top to bottom, the interpolations of $\rho(s)$, $\phi(s)/2\pi$, and $P_\rho(s) - \overline{P_\rho}$ at the time $\bar{s}$. The plot refers to the HNN with the time-splitting approach, for which $\bar{s} = 15.2032$ (vertical dashed line). In the top and bottom panels, we highlight the value 0 with a horizontal dotted line.

at which $\rho(\bar{s}) = 0$. The motion in the interval $[0, \bar{s}]$ corresponds to the incoming phase, in which the particle approaches the fuzzball up to the bounce from it; and, thanks to the symmetry $\rho \mapsto -\rho$ of the system, the outgoing phase can be found by reflecting the incoming motion with respect to $\bar{s}$.

Figure 19 shows the procedure we have employed to determine $\bar{s}$ in the case of HNN. In the top panel, we determine $\bar{s}$ by calculating the time at which the curve $\rho(s)$ intersects the line $\rho = 0$ (assuming a linear *ansatz* due to the small range of times we are focusing on). The resulting time is $\bar{s} = 15.2032$, and it is marked by the red vertical dashed line. The fact that the mathematical solution to the system predicts $\rho < 0$ for $s > \bar{s}$ is irrelevant to the physics,

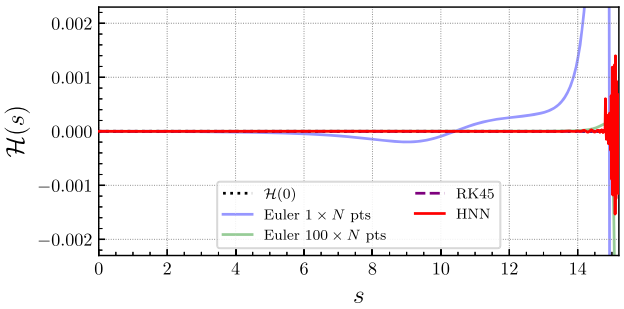

FIG. 20. Energy conservation for the subcritical case with $b = 1.89$ for the different methods, and by following the time-splitting approach. The conserved value of the energy $\mathcal{H}(0) = 0$ is represented by the dotted black line.

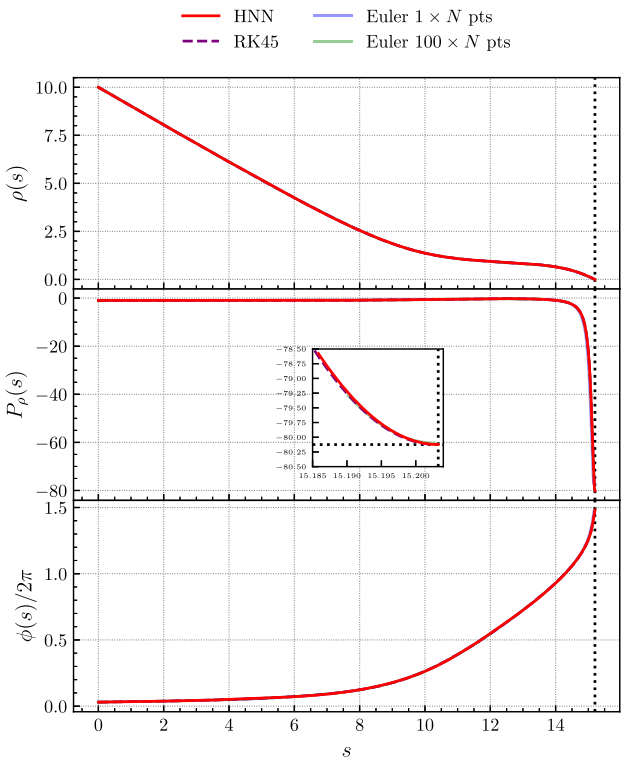

FIG. 21. From top to bottom: $\rho(s)$, $P_\rho(s)$, and $\phi(s)/2\pi$ for the different methods in the subcritical case with $b = 1.89$ and $N = 2400$ obtained with the time-splitting approach. The vertical dotted line in the three panels represents $\bar{s}$ for the HNN. In the middle panel, we also show a zoomed-in view of the region $s \in [15.185, \bar{s}]$.

since the geodesics in the Cartesian plane [see Eq. (4.5)] depends on ρ^2. We then interpolate $\phi(s)/2\pi$ and $P_\rho(s)$ at the time $\bar{s}$ (middle and bottom panels) and discard all the points for which $s > \bar{s}$. The procedure is repeated independently also for the numerical integrators.

The conservation of the energy is shown in Fig. 20 for different methods. The HNN conserves very well the energy over time, and an increase of the oscillations up to ± 0.001 is observed as $s \to \bar{s}$. The Euler method has larger deviations for both N and $100 \times N$ points, while the `RK45` method conserves the energy much better compared to the others.

The dynamical variables $\rho(s)$, $P_\rho(s)$, and $\phi(s)/2\pi$ obtained from the different methods are shown in Fig. 21. No visible differences are observed among them.

To quantify the goodness of the solutions, we still rely on the errors as defined in Eqs. (4.9) and (4.10), where now for the exact $P_\rho(\rho)$ we use Eq. (C7). The errors are plotted in Fig. 22 for the different methods in the incoming motion (the outgoing motion would be identical, since we obtain it from reflection of the incoming motion). The errors made by the HNN tend to increase for both the variables as $\rho \to 0$, but they remain very small compared to the scale of the variables in question, showing that the HNN is able to

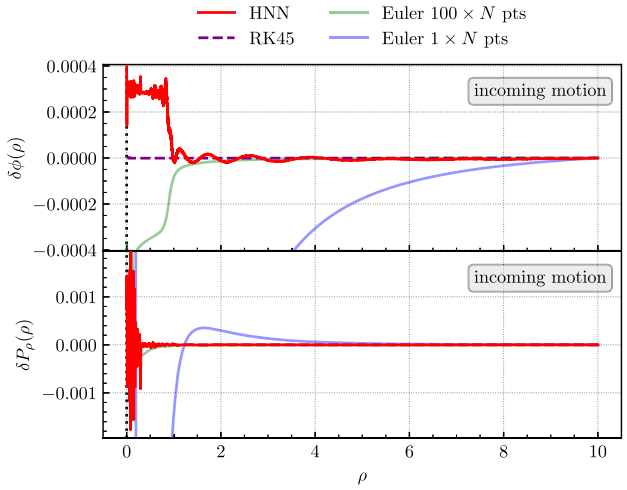

FIG. 22. Errors on the variables $\phi(\rho)$ and $P_\rho(\rho)$ in the subcritical case with $b = 1.89$, obtained from the time-splitting approach.

solve the system with high accuracy. The Euler method with both N and $100 \times N$ points shows larger deviations, while `RK45` is very accurate.

Figure 23 shows the complete (incoming and outgoing phases) geodesics in the Cartesian plane obtained from the different methods. HNN, Euler with $100 \times N$ points, and `RK45` are qualitatively overlapping with the ground truth, while Euler with N points (blue line) deviates significantly from it during the outgoing phase.

2. Approach 2: Preconditioning of the HNN output

The second approach we propose consists in providing an offset to the network's output associated with the problematic variable—in this case, P_ρ—relieving the network from having to learn a particularly stiff behavior. In order to better understand this concept, let us consider the following redefinition:

$$\widehat{P_\rho}(s, \mathbf{w}) = P_{\rho,0} + f(s)[O_{P_\rho}(s, \mathbf{w}) + h(s)], \tag{4.14}$$

where we have introduced the function $h(s)$, representing the new offset with respect to which the output O_{P_ρ} has to be trained to solve the Hamilton equation for $P_\rho(s)$. The function $h(s)$ is chosen arbitrarily and does not depend on the network weights. The idea is to incorporate into $h(s)$ the complexity of the solution $P_\rho(s)$ in such a way that the residual function $O_{P_\rho}(s, \mathbf{w})$, which has to be learned by the HNN, is close to zero or has a much smoother behavior. Let us emphasize that minimizing the loss function leads to finding the solution for $P_\rho(s)$ regardless of the definition of $h(s)$. If $h(s)$ is completely different from the expected solution, the offset will be automatically corrected by the output $O_{P_\rho}(s, \mathbf{w})$. From this perspective, the strategy remains model independent. We will refer to this approach as "preconditioning" of the HNN output.

In order to capture the rapid decrease of $P_\rho(s)$ near $\bar{s}$, we have considered the following Breit-Wigner-type parametrization for $h(s)$:

$$h(s) = \frac{(\overline{P_\rho} - P_{\rho,0})\Gamma^2}{\Gamma^2 + (s - \bar{s})^2}. \tag{4.15}$$

With the aim of minimizing the amount of prior information, we have considered the accurate prediction $\widehat{P_\rho}(s)$ obtained from the time-splitting procedure of the previous subsection, then $\bar{s} = 15.2032$ and fitted the quantity $\widehat{P_\rho}(s) - P_{\rho,0}$ in the time range $s \in [14, \bar{s}]$ with the fit *ansatz* (4.15), getting $\Gamma = 0.1213$. In the fitting procedure, we have neglected the effect of the function $f(s)$, since we are interested in parametrizing the solution at $s \gg 1$, for which $f(s) \simeq 1$. The result of the fit is drawn in blue in Fig. 24. As can be seen, the so-defined $h(s)$ function

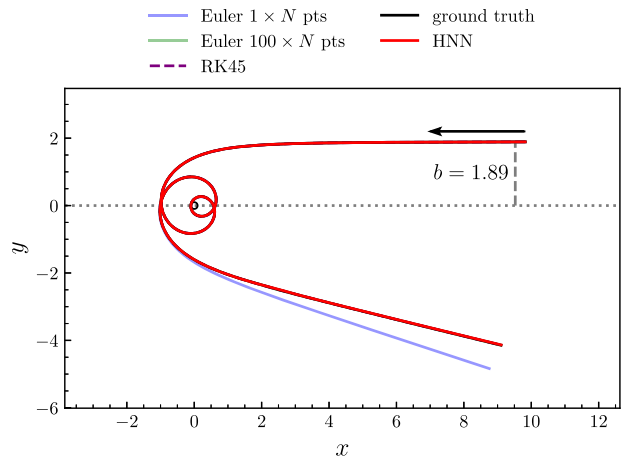

FIG. 23. Geodesics in the Cartesian plane obtained from different methods in the subcritical case with $b = 1.89$ and by applying the time-splitting approach. The ground truth is represented by the solid black line, while the small circle at the center (0,0) has radius a_f, and it represents the fuzzball. The black arrow indicates the incoming direction of the particle.

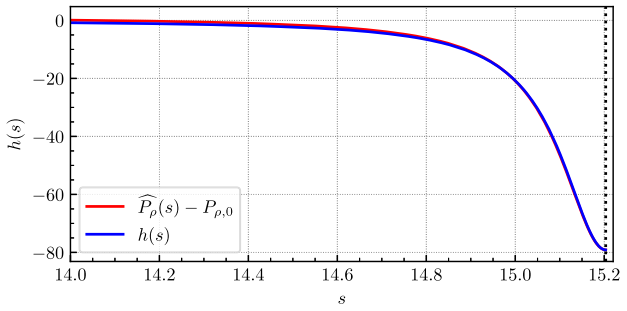

FIG. 24. Result of the fit (blue line) compared to the fitted quantity (red line). In this case, $\widehat{P_\rho}(s)$ is the prediction of the HNN obtained from the time-splitting approach in Sec. IV C 1.

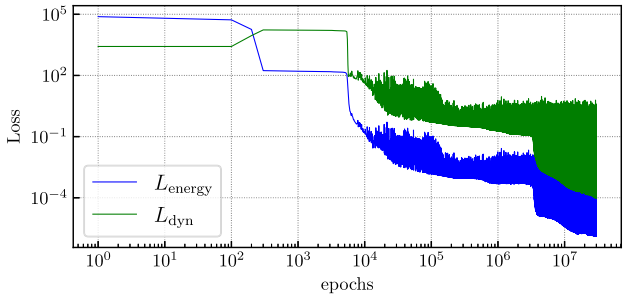

FIG. 25. Loss functions for the subcritical case with $b = 1.89$ by using the preconditioning approach.

provides an excellent parametrization for $\widehat{P_\rho}(s) - P_{\rho,0}$ (in red) for times close to $\bar{s}$. Let us say that in general, one can rely on other sources of information, if available, such as the result of the numerical integrators, or one can design *Ansätze* by leveraging theoretical knowledge of the system.

With this setup, we have trained a neural network made of three hidden layers and 64 neurons for each for 30×10^6 epochs. η_i, appearing in Eq. (4.4), is set to 4×10^{-4}. The time interval is $\mathcal{T}_{\text{train}} = [0, 15.22]$, and it is divided into $N = 800$ points. In addition, we have set $\gamma_{P_\rho} = 0.01$ [recall the definition of the dynamics loss function in Eq. (2.11)] in order to give more importance to the dynamical equations for $\rho(s)$ and $\phi(s)$, since the one for $P_\rho(s)$ is basically already solved.

The loss functions are shown in Fig. 25 and are at a similar level to those corresponding to the third interval in the time-splitting approach, which are displayed in the bottom panel of Fig. 18.

The results from the numerical integrators are taken from the previous subsection, and we will not comment on them here again. For the HNN predictions, we have repeated the interpolation procedure illustrated in Fig. 19 to determine $\bar{s}$ (which still remains at the value 15.2032), and consequently, to discard the points for which $s > \bar{s}$. This procedure is also needed to obtain the outgoing trajectory by reflection of the incoming one.

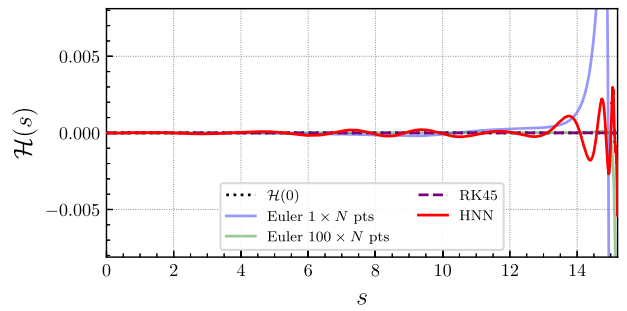

FIG. 26. Energy conservation for the subcritical case with $b = 1.89$ for the different methods and by following the preconditioning approach. The conserved value of the energy $\mathcal{H}(0) = 0$ is represented by the dotted black line.

As shown in Fig. 26, the energy is well conserved on average and, compared to the time-splitting method (see Fig. 20), the deviations from $\mathcal{H}(0) = 0$ are slightly larger and reach 0.005 in absolute value for $s \to \bar{s}$. It should be noted that both L_{dyn} and L_{energy} have not yet reached a plateau, meaning that the accuracy of the solution, and then the conservation of the energy, can be further improved by training the network for more epochs.

The predicted variables $\rho(s)$, $P_\rho(s)$, and $\phi(s)/2\pi$ are shown in Fig. 27 from top to bottom. Analogously to what is observed in the time-splitting approach (Fig. 21), there is no visually appreciable difference between the different methods.

In order to make a quantitative statement, we plot the errors $\delta\phi(\rho)$ and $\delta P_\rho(\rho)$ in the top and bottom panels of Fig. 28, respectively. For $\phi(\rho)$, we observe errors that are about 100 times larger than than those shown in the top panel of Fig. 22, while for P_ρ we see that the error increases by about a factor of 5. The worse performance compared to before can be improved by increasing the number of epochs, but we emphasize that the error scale remains very small.

Finally, in Fig. 29, we show the predicted geodesic in the Cartesian plane compared to the other methods and to the

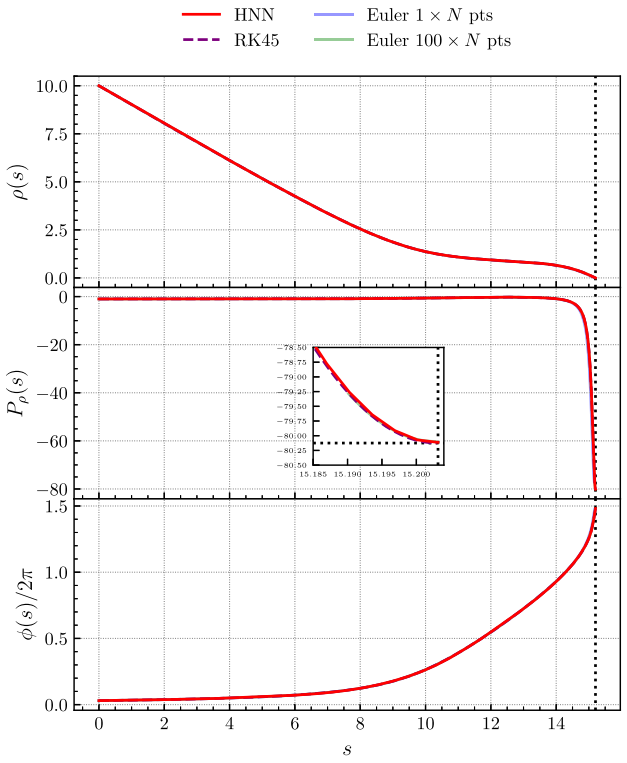

FIG. 27. From top to bottom: $\rho(s)$, $P_\rho(s)$, and $\phi(s)/2\pi$ for the different methods in the subcritical case with $b = 1.89$ and $N = 2400$ obtained with the preconditioning approach. The vertical dotted line in the three panels represents $\bar{s}$ for the HNN. In the middle panel, we also show a zoomed-in view of the region $s \in [15.185, \bar{s}]$.

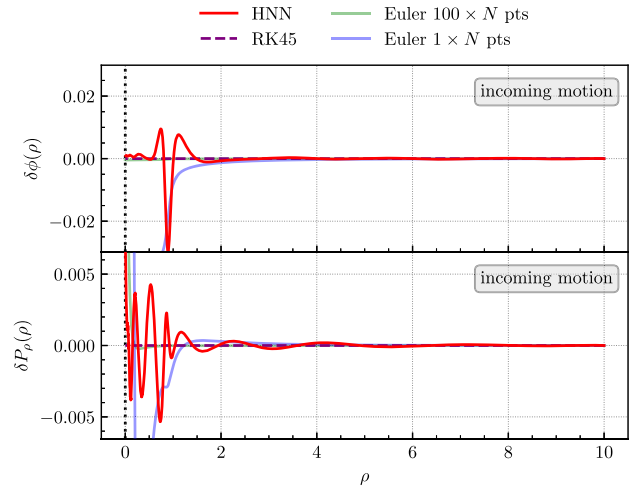

FIG. 28. Errors on the variables $\phi(\rho)$ and $P_\rho(\rho)$ in the subcritical case with $b = 1.89$ obtained from the preconditioning approach.

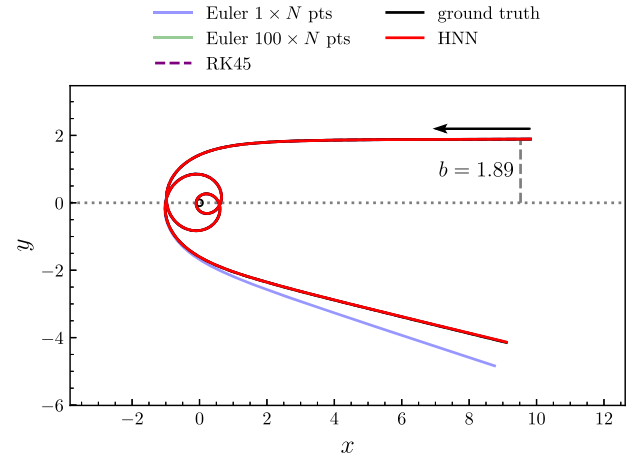

FIG. 29. Geodesics in the Cartesian plane obtained from different methods in the subcritical case with $b = 1.89$ and by applying the preconditioning approach. The ground truth is represented by the solid black line, while the small circle at the center (0,0) has radius a_f and represents the fuzzball. The black arrow indicates the incoming direction of the particle.

ground truth. As we can see, the solution predicted by the HNN is still very accurate and in agreement with the ground truth. The strategy proposed here effectively allows incorporating information about the solutions during the training phase while preserving model independence and facilitating the minimization process of the loss function. We leave the exploration of further approaches for future work. With this, we conclude the results concerning the planar geodesics.

V. RESULTS FOR NONPLANAR GEODESICS

In this section, we present the results for the nonplanar case which is discussed in Sec. III E and whose dynamics is determined by the following five (J_ϕ is not considered, since it is constant) Hamilton equations:

$$\dot{\rho} = \frac{\partial \mathcal{H}}{\partial P_\rho}, \qquad \dot{P}_\rho = -\frac{\partial \mathcal{H}}{\partial \rho}, \qquad \dot{\phi} = \frac{\partial \mathcal{H}}{\partial J_\phi},$$
$$\dot{\theta} = \frac{\partial \mathcal{H}}{\partial P_\theta}, \qquad \dot{P}_\theta = -\frac{\partial \mathcal{H}}{\partial \theta}. \tag{5.1}$$

The derivatives on the right-hand side are explicitly written in Appendix A 2, and one can immediately appreciate the extreme complexity of such a set of equations. Indeed, the nonplanar case presents a much greater challenge compared to the planar one, due to the higher number of coupled dynamical variables and the increased complexity of the dynamics.

As already hinted in Sec. III E, our analysis of nonplanar geodesics has been limited to the critical ones, as they are crucial from both an observational and a theoretical perspective. Their determination relies on the separability of motion. However, when separability does not hold, determining critical geodesics becomes challenging, and we leave a detailed investigation of this case to future work.

The geometry parameters that we fix in this analysis are $L_1 = 0.5$, $L_5 = 1$, $a_f = 0.2$, $E = 1$ (from Appendix D, it is clear that L_1 and L_5 cannot be equal). With this choice, the critical parameters ζ_c, b_c, and ρ_c assume the values

$$\zeta_c = -2.5, \tag{5.2}$$

$$b_c = 1.5, \tag{5.3}$$

$$\rho_c = \sqrt{0.46} = 0.6782, \tag{5.4}$$

while $b_{\phi,c} = 0$ (see Appendix D).

In order to accommodate the Hamilton equations (5.1), we need a neural network with five output nodes, from which the solutions are assembled according to

$$\hat{\rho}(s) = \rho_0 + f(s)O_\rho(s),$$
$$\widehat{P_\rho}(s) = P_{\rho,0} + f(s)O_{P_\rho}(s),$$
$$\hat{\phi}(s) = \phi_0 + f(s)O_\phi(s),$$
$$\widehat{P_\theta}(s) = P_{\theta,0} + f(s)O_{P_\theta}(s),$$
$$\hat{\theta}(s) = \theta_0 + f(s)O_\theta(s). \tag{5.5}$$

We have chosen the following initial conditions:

$$\rho_0 = 3, \qquad \theta_0 = 0.9, \qquad \phi_0 = 1. \tag{5.6}$$

The value $P_{\theta,0}$ is obtained from Eq. (3.25); in particular, we have (recall that $E = 1$)

$$\frac{P_{\theta,0}^2}{E^2} = P_{\theta,0}^2 = b_c^2 - a_f^2 \sin^2\theta_0, \tag{5.7}$$

from which

$$P_{\theta,0} = 1.4918. \tag{5.8}$$

The positive sign of the solution determines the initial direction of the trajectory. Finally, $P_{\rho,0}$ is obtained by requiring that the Hamiltonian in Eq. (3.23) vanish at $s = 0$. Using Eqs. (5.2), (5.6), and (5.8), we gain

$$P_{\rho,0} = -0.9447. \tag{5.9}$$

Again, we will compare the prediction of the HNN with the `RK45` method and with the symplectic first-order Euler method. However, in this case, we are unable to derive a ground truth for the geodesic in the Cartesian space, and therefore we cannot quantify the errors introduced by the different integration methods. To assess the quality of a solution, we can only rely on the expected theoretical behavior of the variables.

Since we have chosen the parameters to select the critical trajectory, $\rho(s)$ is expected to decrease from ρ_0 until it reaches and settles at the value ρ_c. Consequently, $P_\rho(s)$ is expected to increase from the negative $P_{\rho,0}$ to 0 and remain at that value. Concerning $\theta(s)$ and $\phi(s)$, we expect that, when the particle enters the critical regime with $\rho(s) \simeq \rho_c$ and it wraps around the oblate spheroid [Eq. (D8)], they show linear growth in time. Consequently, $P_\theta(s)$ in the critical regime should have an oscillating trend with constant period.

For completeness, we project the oblate spheroid in Eq. (D8) onto the three planes x-y ($\theta = \pi/2$), x-z ($\phi = 0$), and y-z ($\phi = \pi/2$). To do this, we rely on Eq. (3.22), which maps the spherical coordinates ρ, θ, and ϕ to the Cartesian ones x, y, and z. It turns out that the profile of the geodesic in the plane x-y is given by the circumference

$$x^2 + y^2 = \rho_c^2 + a_f^2 \qquad x\text{-}y \text{ plane}, \tag{5.10}$$

and, in the other two planes, by the following two ellipses:

$$\frac{x^2}{\rho_c^2 + a_f^2} + \frac{z^2}{\rho_c^2} = 1 \qquad x\text{-}z \text{ plane}, \tag{5.11}$$

$$\frac{y^2}{\rho_c^2 + a_f^2} + \frac{z^2}{\rho_c^2} = 1 \qquad y\text{-}z \text{ plane}. \tag{5.12}$$

This concludes our *a priori* knowledge of the geodesic, based on physical principles.

We now discuss the training phase of the neural network. We have thoroughly investigated various parameter choices and architectures. Below, we present the strategy that, according to our analysis, has proven to be the most efficient. As the training time interval, we have set the range $\mathcal{T}_{\text{train}} = [0, 60]$, equally sampled with $N = 2400$ points. Also in this case, after the training, we have validated the loss function on the interval $\mathcal{T}_{\text{val}}$ to check that no overfitting has occurred.

The first difference with respect to the planar case is the choice of the neural network architecture. As anticipated, the variable $P_\theta(s)$ is expected to have a periodic trend, in contrast to the monotonic trend of the others. For this reason, we have found it convenient to consider a two-block architecture with a common input node given by the time. The first block consists of two hidden layers with 64 neurons in each and the output O_{P_θ}. For this first block, we have used a sinusoidal activation function since, as remarked in Ref. [2], using trigonometric functions is particularly convenient when the output is expected to have a periodic behavior. The second block consists of two hidden layers with 128 neurons in each, a hyperbolic-tangent activation function, and the remaining four outputs O_ρ, O_{P_ρ}, O_ϕ, and O_θ.

The second difference is the addition of a penalty term to the loss function (2.15). This contribution is introduced to penalize the solutions such that $P_\rho(s) > 0$ and takes the form

$$L_{\text{penalty}}(\mathbf{w}) = \sum_{s \in \mathcal{S}_p} [\max(\widehat{P_\rho}(s, \mathbf{w}), 0)]^2, \tag{5.13}$$

where $\mathcal{S}_p = \{8, 20, 30, 40, 50, 60\}$. The total loss function actually minimized is therefore

$$L(\lambda, \mathbf{w}) = (1 - \lambda) L_{\text{dyn}}(\mathbf{w}) + \lambda L_{\text{energy}}(\mathbf{w}) + L_{\text{penalty}}(\mathbf{w}). \tag{5.14}$$

The penalty term is activated only for $\widehat{P_\rho}(s, \mathbf{w}) > 0$, while it does not contribute to the loss function when, correctly, $\widehat{P_\rho}(s, \mathbf{w}) < 0$. We have observed that without this additional term, the HNN tends to predict unstable trajectories. Let us stress that we are using information from the physics that can be deduced without any prior knowledge of the actual solution. This is a clear example of how various types of information about the solutions can be fruitfully incorporated into the loss function to restrict the solution space and improve the network's predictability. The training is performed by using the Adam optimizer, and we monitor individually both L_{dyn} and L_{energy}.

Concerning the remaining algorithmic parameters, the ones entering $\eta(\text{epoch})$ in Eq. (4.4) are $\eta_i = 8 \times 10^{-8}$, $\eta_f = \eta_i/50$, $\eta_c = 500 \times 10^3$, and $\sigma_\eta = 100 \times 10^3$. In this case, it turns out that the small value $\lambda = 0.1$ gives better performances compared to $\lambda = 0.99$ used in all the cases of the planar study. We interpret this result in light of the fact that the dynamics are much more complex, and therefore

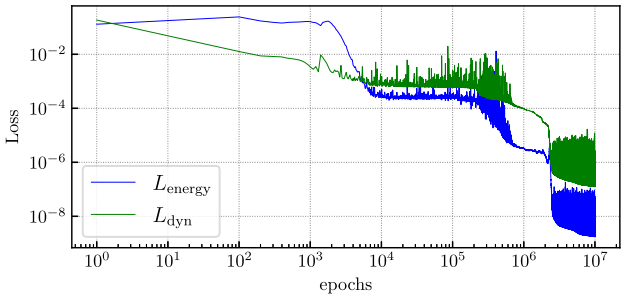

FIG. 30. Loss functions for the nonplanar case.

should be prioritized over energy conservation during the minimization process.

Having fixed the setup, we have trained the neural network with 10×10^6 epochs, and the loss functions are shown in Fig. 30. The plot shows a good performance, which is comparable to those obtained in the planar case.

In Fig. 31, we show the energy $\mathcal{H}(s)$ for the different methods. The HNN conserves the energy on average with small fluctuations between -0.0001 and $+0.0001$. The Euler methods start failing in conserving the energy already at small times. The `RK45` method appears to conserve energy with great accuracy, but, as we have already seen in the planar critical case in Sec. IV B, this does not necessarily mean that its numerical solution is the correct one.

We realize this by looking at the predicted $\hat{\rho}(s)$, $\widehat{P_\rho}(s)$, $\hat{\phi}(s)/2\pi$, $\widehat{P_\theta}(s)$, and $\hat{\theta}(s)/\pi$, which are shown, from top to bottom in the given order, in Fig. 32. As can be seen, the different methods yield very different solutions, and the one predicted by the HNN is the only one which is fully in agreement with the theoretical expectations discussed above. The critical regime for which $\rho(s) \simeq \rho_c$ is reached around $s = 6$, and it is maintained stable for all subsequent times. The prediction of $P_\theta(s)$ as an oscillating function is remarkably good, and such a result is due to the use of a trigonometric activation function in the architecture block responsible for the output associated with P_θ. The Euler method, both with N and with $100 \times N$ points, fails to

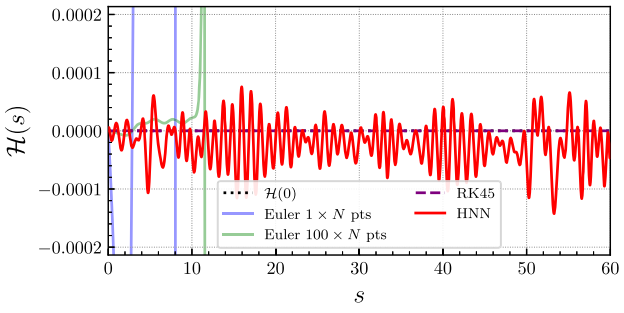

FIG. 31. Energy conservation over the time for the nonplanar case. The conserved value of the energy $\mathcal{H}(0) = 0$ is represented by the dotted black line.

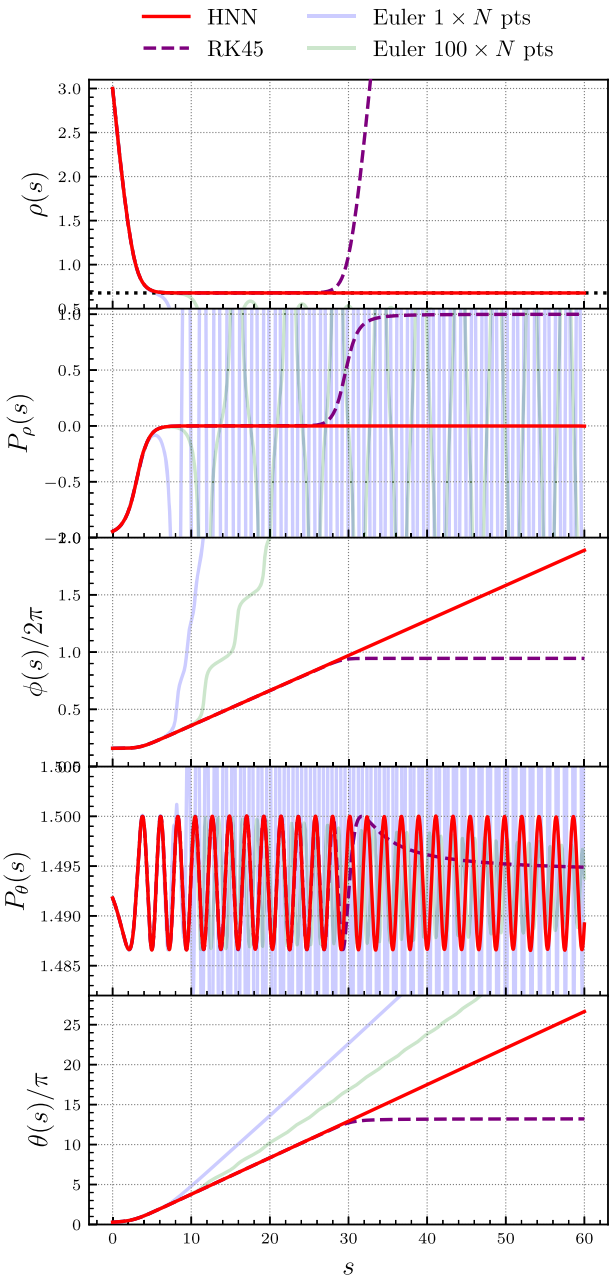

FIG. 32. From top to bottom: $\rho(s)$, $P_\rho(s)$, $\phi(s)/2\pi$, $P_\theta(s)$, and $\theta(s)/\pi$ for the different methods in the nonplanar case. The horizontal black dotted line in the top panel represents the critical radius $\rho_c = 0.6782$, on which the trajectory must settle. The trajectory predicted by the HNN is the only one among the different methods to correctly reproduce this feature.

remain in the critical regime and even predicts $\rho(s) \simeq 0$, which corresponds to a particle which has collapsed onto the fuzzball. The `RK45` method is in agreement with the HNN up to $s \simeq 28$, and after this, $\rho(s)$ starts increasing, while the angular variables flatten, showing a particle that moves away from the critical trajectory as if it were in a situation analogous to the overcritical case presented in Sec. IV A.

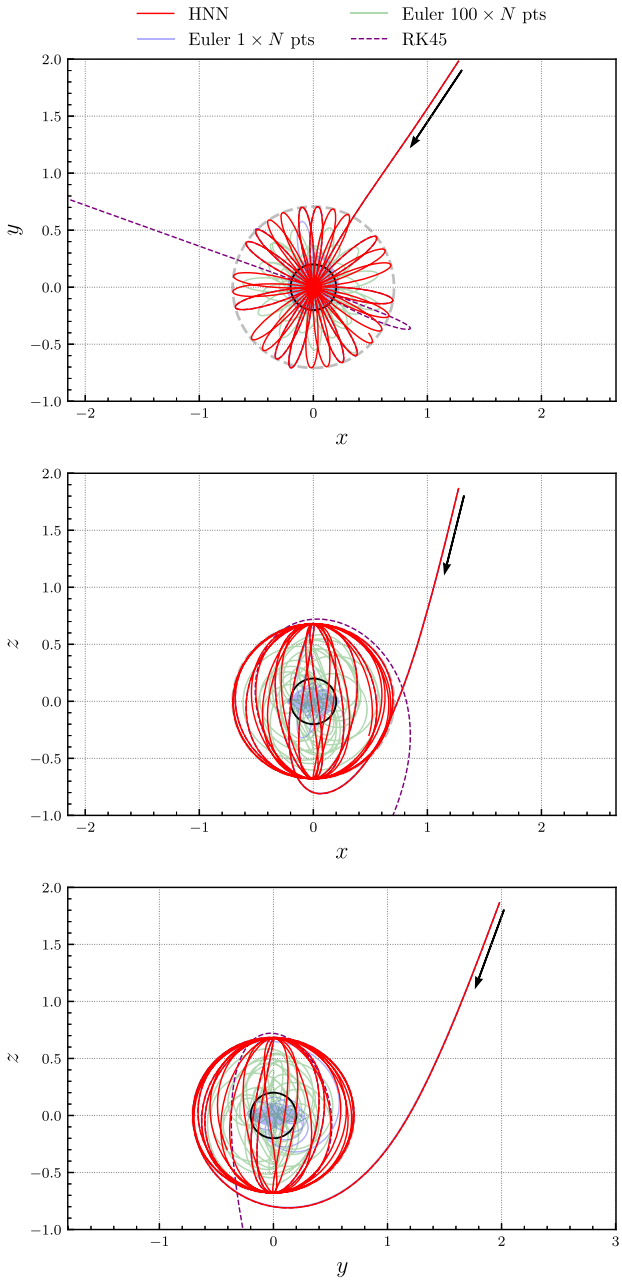

FIG. 33. Geodesics in the nonplanar case for different methods, projected onto the x-y, x-z, and y-z Cartesian planes, respectively, from top to bottom. The solid black line represents the fuzzball surface. The dashed light-gray lines represent the circumference and ellipses defined in Eqs. (5.10)–(5.12). The black arrow indicates the incoming direction of the particle.

We now convert the spherical coordinates to the Cartesian ones by using Eq. (3.22). In Fig. 33, we show the trajectories from the different methods on the x-y, x-z, and y-z planes, from top to bottom, respectively. We also display the curves (colored in gray) given in Eqs. (5.10)–(5.12), which represent the profile of the oblate spheroid. We can appreciate how the geodesic predicted by the HNN, once entered in the critical regime, exactly describes the

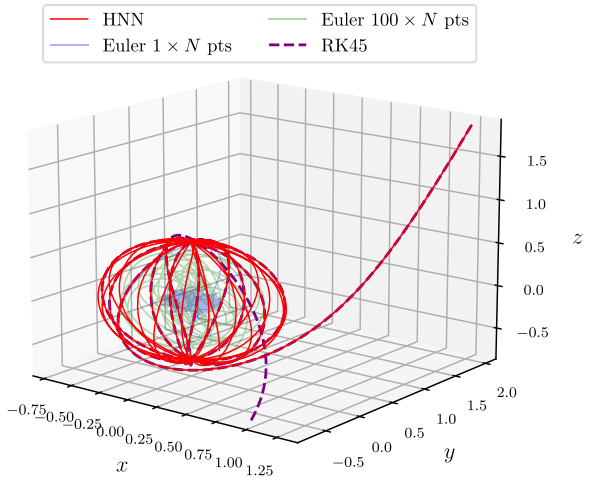

FIG. 34. Geodesics in the 3D Cartesian space for different methods in the nonplanar geometry.

profile given by the aforementioned curves in the three Cartesian planes. With this observation, we can conclude that the HNN is the only method that, in this challenging situation due to the instability of the critical solution, solves the Hamilton equations with high accuracy and allows for a solid and stable numerical study of the critical nonplanar geodesic of a massless particle immersed in the D1-D5 circular fuzzball. This is the main result of this paper and paves the way for future studies on more complex geometries.

We conclude this section by showing the geodesics in the 3D Cartesian space in Fig. 34. While it does not add anything new to the previous discussion, it still looks beautiful and, frankly, quite promising.

VI. CONCLUSIONS AND OUTLOOKS

Inspired by Ref. [2], we studied the massless and neutral geodesics in a peculiar, smooth, and horizonless fuzzball geometry as an innovative application of Hamiltonian neural networks (HNNs). For this physical system, extensively studied in Refs. [37–39], it is possible to obtain an explicit ground truth, which has allowed for a comparative study of the performance of HNNs and other numerical integrators, specifically the first-order semi-implicit Euler method and the `RK45` method. The former is symplectic, while the latter is not but allows for setting bounds on the relative and absolute errors on the solution. We have analyzed subcritical, critical, and overcritical planar geodesics, as well as nonplanar critical geodesics. The critical ones that are nontrivial are unstable, making it challenging to obtain accurate results using standard numerical integrators, as they are known to accumulate errors over time.

Our analysis has shown that, despite the indisputable complexity of the equations learned, which can be appreciated in Appendix A, HNNs were able to accurately reproduce the geodesics across all the mentioned cases,

even over large timescales, whereas the considered numerical integrators failed, as expected, when computing unstable trajectories. This result highlights another significant aspect that emerged from our analysis: the performance of HNNs, due to the fact that the solution is obtained by simultaneously and independently considering all time steps, does not depend on the intrinsic stability or instability of the dynamics described by the Hamilton equations, and it remains at the same level for the different cases. In contrast, the performance of numerical integrators is highly case dependent. In order to complete the comparison, we emphasize that HNNs are model independent and produce an analytic time-dependent output, whereas numerical integrators predict the solution at a finite and discrete set of time steps. Additionally, HNNs allow for easy incorporation of constraints, symmetries and data-driven information to enhance the solution's predictability as is done, for example, in the nonplanar case. The set of theorems and statements collectively known as the universal approximation theorem [43,44] further guarantees that neural networks, as function approximators, are always improvable in terms of accuracy. All these advantages make HNNs not only an alternative, but a preferable tool in terms of flexibility and robustness compared to standard numerical integrators. This remains true even in cases where the `RK45` method appears to produce more precise results. In fact, as observed in the critical cases, the nonsymplectic nature of `RK45` can lead to solutions that exactly conserve energy but are nonetheless incorrect. On the other hand, a symplectic method like the Euler approach often requires an infinitesimally small time step—and consequently, an immense number of iterations—to produce accurate solutions.

The results and evidence gathered in this paper are clearly generalizable to other physical systems, since the concept of stability and instability extends far beyond the case of a particle moving in a fuzzball geometry. Our work thus contributes to the already flourishing field of ML and PINN applications across various domains of physics and science more broadly.

However, in our view, a major downside of ML remains the fact that implementing a neural network and tuning its parameters to find the optimal setup is a significantly more laborious and time-consuming process compared to applying standard methods, which are often readily available as default packages in common programming languages. Nevertheless, this issue can be mitigated by the numerous libraries and dedicated tools that have made neural network implementation accessible and straightforward, even on standard desktop notebooks.

For the authors, this work represents the initial step in a long-term project, aimed at systematically studying geometries emerging in string theory through HNNs and PINNs. For instance, as a future outlook, a detailed study of geodesic motion in the so-called D1-D5-P (three-charge) fuzzball would be highly interesting. This is a nontrivial generalization of the present D1-D5 (two-charge) fuzzball with an additional Kaluza-Klein (KK) charge. The main complication arises from the fact that, in general, this geometry does not allow for a separation of radial and angular dynamics via the introduction of a Carter-like constant; separation is possible only on the equatorial planes [60]. After the detailed experience with the separable two-charge case, we plan to apply the HNN to this more complex physical system, which, due to its considerable intricacy, has never been studied in terms of either geodesic motion or wave propagation.

In this case, there is no ground truth for comparison, and a challenging aspect we intend to address is finding a way to assess the quality of the prediction. This issue, as discussed above, also applies to numerical integrators. For HNNs, we can first rely on the loss function itself, which provides a quantitative measure of how well the predicted solution satisfies the equations of motion. Second, multiple optimal network setups may exist, and studying the stability of the predicted solution under variations in algorithmic parameters—such as the number of time steps, the length of the time interval, the choice of activation functions, and the network architecture—serves as both a qualitative and quantitative indicator of accuracy. We plan to further explore and refine this type of analysis in our next work.

Finally, studying the geodesic motion in this new scenario will help to identify the photon spheres, which serve as a polar star in analyzing the quasinormal mode (QNM) spectrum [40–42], an investigation entrusted to a future stage of the project.

ACKNOWLEDGMENTS

We are grateful to N. Tantalo for his illuminating comments on a preliminary version of the manuscript. We warmly thank M. Bianchi for the discussion that led to the beginning of this work and for useful comments. We also want to thank the PRIN "String Theory as a Bridge between Gauge Theory and Quantum Gravity" for having created a stimulating environment. A. C. would like to thank the Department of Theoretical Physics at CERN for kind hospitality during the final stages of the present work.

DATA AVAILABILITY

The data that support the findings of this article are not publicly available upon publication but are available from the authors upon request.

APPENDIX A: HAMILTON EQUATIONS

In this appendix, we report the derivatives of the Hamiltonian in the planar and nonplanar cases that are used inside the loss functions. Due to the huge length of these relations, symbolic writing will be used, and that will be specified.

1. Planar case

From the Hamiltonian (3.13), the derivatives to be considered are

$$
\begin{aligned}
-\dot{P}_\rho = \frac{\partial \mathcal{H}}{\partial \rho} &= \frac{E^2[-H_1(-4H_5\omega_\phi((a_f^2+\rho^2)\omega_{\phi,\rho}-\rho\omega_\phi)+\omega_\phi^2 H_{5,\rho}(a_f^2+\rho^2)+H_5^2 H_{1,\rho}(a_f^2+\rho^2)^2)]}{4H_1^{3/2}H_5^{3/2}(a_f^2+\rho^2)^2} \\
&+\frac{-E^2(H_5\omega_\phi^2 H_{1,\rho}(a_f^2+\rho^2)+H_5H_1^2H_{5,\rho}(-(a_f^2+\rho^2)^2))}{4H_1^{3/2}H_5^{3/2}(a_f^2+\rho^2)^2} \\
&+\frac{EJ_\phi(H_1(2H_5((a_f^2+\rho^2)\omega_{\phi,\rho}-2\rho\omega_\phi)-\omega_\phi H_{5,\rho}(a_f^2+\rho^2))-H_5\omega_\phi H_{1,\rho}(a_f^2+\rho^2))}{2H_1^{3/2}H_5^{3/2}(a_f^2+\rho^2)^2} \\
&+\frac{J_\phi^2(-(H_5H_{1,\rho}(a_f^2+\rho^2))-H_1(H_{5,\rho}(a_f^2+\rho^2)+4H_5\rho))}{4H_1^{3/2}H_5^{3/2}(a_f^2+\rho^2)^2} \\
&+\frac{P_\rho^2(H_5(-\chi)H_{1,\rho}(a_f^2+\rho^2)-H_1(\chi H_{5,\rho}(a_f^2+\rho^2)+4H_5(\chi_\rho(a_f^2+\rho^2)-\rho\chi)))}{4H_1^{3/2}H_5^{3/2}\chi^3},
\end{aligned}
\tag{A1}
$$

$$
\dot{\rho} = \frac{\partial \mathcal{H}}{\partial P_\rho} = \frac{P_\rho(a_f^2+\rho^2)}{\sqrt{H_1}\sqrt{H_5}\chi^2}, \tag{A2}
$$

$$
-\dot{J_\phi} = \frac{\partial \mathcal{H}}{\partial \phi} = 0, \tag{A3}
$$

$$
\dot{\phi} = \frac{\partial \mathcal{H}}{\partial J_\phi} = \frac{E\omega_\phi + J_\phi}{\sqrt{H_1}\sqrt{H_5}(a_f^2+\rho^2)}, \tag{A4}
$$

where we have adopted the following conventions:

(1) $\chi^2 \equiv \chi^2(\rho,\theta) = \rho^2 + a_f^2\cos^2\theta$.

(2) All the quantities that depend on θ (such as $H_1, H_5, \chi, \omega_\phi$) are evaluated at $\theta = \pi/2$.

(3) $f_{,x} \equiv \partial f/\partial x$ for any quantity f, except for $\chi_\rho = \partial\chi/\partial\rho$. These derivatives are evaluated in $\theta = \pi/2$ as well.

Notice that Eq. (A3) is due to the fact that ϕ is cyclic, and consequently the corresponding conjugate momentum is conserved. The functions H_1 and H_5 have already been defined in Eq. (3.7).

2. Nonplanar case

From the Hamiltonian (3.23), the derivatives to be considered are

$$
\begin{aligned}
-\dot{P}_\rho = \frac{\partial \mathcal{H}}{\partial \rho} &= \frac{E^2(H_1(H_5^2(c_\theta^2-1)H_{1,\rho}(-(a_f^2+\rho^2)^2)-4H_5\omega_\phi((a_f^2+\rho^2)\omega_{\phi,\rho}-\rho\omega_\phi)+\omega_\phi^2 H_{5,\rho}(a_f^2+\rho^2)))}{4H_1^{3/2}H_5^{3/2}(c_\theta^2-1)(a_f^2+\rho^2)^2} \\
&+\frac{E^2(H_5H_1^2(c_\theta^2-1)H_{5,\rho}(-(a_f^2+\rho^2)^2)+H_5\omega_\phi^2 H_{1,\rho}(a_f^2+\rho^2))}{4H_1^{3/2}H_5^{3/2}(c_\theta^2-1)(a_f^2+\rho^2)^2}+\frac{P_\theta^2(-H_5(\chi H_{1,\rho}+4H_1\chi_\rho)-H_1\chi H_{5,\rho})}{4H_1^{3/2}H_5^{3/2}\chi^3} \\
&+\frac{P_\rho^2(H_5(-\chi)H_{1,\rho}(a_f^2+\rho^2)-H_1(\chi H_{5,\rho}(a_f^2+\rho^2)+4H_5(\chi_\rho(a_f^2+\rho^2)-\rho\chi)))}{4H_1^{3/2}H_5^{3/2}\chi^3} \\
&+\frac{EJ_\phi(H_5\omega_\phi H_{1,\rho}(a_f^2+\rho^2)+H_1(\omega_\phi H_{5,\rho}(a_f^2+\rho^2)-2H_5((a_f^2+\rho^2)\omega_{\phi,\rho}-2\rho\omega_\phi)))}{2H_1^{3/2}H_5^{3/2}(c_\theta^2-1)(a_f^2+\rho^2)^2} \\
&+\frac{J_\phi^2(H_5H_{1,\rho}(a_f^2+\rho^2)+H_1(H_{5,\rho}(a_f^2+\rho^2)+4H_5\rho))}{4H_1^{3/2}H_5^{3/2}(c_\theta^2-1)(a_f^2+\rho^2)^2},
\end{aligned}
\tag{A5}
$$

$$\dot{\rho} = \frac{\partial \mathcal{H}}{\partial P_\rho} = \frac{P_\rho (a_f^2 + \rho^2)}{\sqrt{H_1}\sqrt{H_5}\chi^2}, \tag{A6}$$

$$-\dot{J}_\phi = \frac{\partial \mathcal{H}}{\partial \phi} = 0, \tag{A7}$$

$$\dot{\phi} = \frac{\partial \mathcal{H}}{\partial J_\phi} = \frac{\csc^2(\theta)(E\omega_\phi + J_\phi)}{\sqrt{H_1}\sqrt{H_5}(a_f^2 + \rho^2)}, \tag{A8}$$

$$\dot{\theta} = \frac{\partial \mathcal{H}}{\partial P_\theta} = \frac{P_\theta}{\sqrt{H_1}\sqrt{H_5}\chi^2}, \tag{A9}$$

$$\begin{aligned}
-\dot{P}_\theta = \frac{\partial \mathcal{H}}{\partial \theta} = &-\frac{E^2(H_1(H_5^2 s_\theta^3 H_{1,\theta}(a_f^2+\rho^2) + 4H_5\omega_\phi(c_\theta\omega_\phi - s_\theta\omega_{\phi,\theta}) + s_\theta\omega_\phi^2 H_{5,\theta}) + H_1^2 H_5 s_\theta^3 H_{5,\theta}(a_f^2+\rho^2))}{4H_1^{3/2}H_5^{3/2}s_\theta^3(a_f^2+\rho^2)} \\
&- \frac{E^2(H_5 s_\theta \omega_\phi^2 H_{1,\theta})}{4H_1^{3/2}H_5^{3/2}s_\theta^3(a_f^2+\rho^2)} - \frac{EJ_\phi(H_1(H_5(4c_\theta\omega_\phi - 2s_\theta\omega_{\phi,\theta}) + s_\theta\omega_\phi H_{5,\theta}) + H_5 s_\theta \omega_\phi H_{1,\theta})}{2H_1^{3/2}H_5^{3/2}s_\theta^3(a_f^2+\rho^2)} \\
&+ \frac{P_\theta^2(-H_5(\chi H_{1,\theta} + 4H_1\chi_\theta) - H_1\chi H_{5,\theta})}{4H_1^{3/2}H_5^{3/2}\chi^3} - \frac{P_\rho^2(a_f^2+\rho^2)(H_5(\chi H_{1,\theta} + 4H_1\chi_\theta) + H_1\chi H_{5,\theta})}{4H_1^{3/2}H_5^{3/2}\chi^3} \\
&- \frac{J_\phi^2(H_1(s_\theta H_{5,\theta} + 4H_5 c_\theta) + H_5 s_\theta H_{1,\theta})}{4H_1^{3/2}H_5^{3/2}s_\theta^3(a_f^2+\rho^2)},
\end{aligned} \tag{A10}$$

where the conventions are the same as for the planar case, with the only difference being that now the quantities are no longer evaluated anymore in $\theta = \pi/2$, but they are functions of θ. Furthermore, we have introduced the symbols $c_\theta = \cos\theta$, $s_\theta = \sin\theta$.

APPENDIX B: SEPARABILITY

In this appendix, we report the details that allow us to determine the Eq. (3.17) of the separable case. We start by computing the conjugate momenta using Eq. (3.4), knowing the Lagrangian in the more general case (meaning keeping in count also the coordinates z_I, with $I = 1, \ldots, 4$, that have been neglected throughout all the work). We get the following results:

$$\begin{aligned}
&P_t = -\frac{\dot{t} + \omega_\phi\dot{\phi}}{H} = -E, \qquad P_z = \frac{\dot{z} + \omega_\psi\dot{\psi}}{H}, \qquad P_\rho = \frac{H(\rho^2 + a_f^2\cos^2\theta)}{\rho^2 + a_f^2}\dot{\rho}, \qquad P_\theta = H(\rho^2 + a_f^2\cos^2\theta)\dot{\theta}, \\
&P_\phi = -\frac{\omega_\phi}{H}\dot{t} + \left[H(\rho^2 + a_f^2)\sin^2\theta - \frac{\omega_\phi^2}{H}\right]\dot{\phi} = J_\phi, \qquad P_\psi = \frac{\omega_\psi}{H}\dot{z} + \left(H\rho^2\cos^2\theta + \frac{\omega_\psi^2}{H}\right)\dot{\psi} = J_\psi, \\
&P_{1,2} = \left(\frac{H_1}{H_5}\right)^{1/2}\dot{z}_{1,2}, \qquad P_{3,4} = \left(\frac{H_5}{H_1}\right)^{1/2}\dot{z}_{3,4},
\end{aligned} \tag{B1}$$

where the only nonconserved momenta are P_ρ and P_θ. By inverting the previous relations, we obtain

$$\begin{aligned}
&\dot{t} = HE - \frac{\omega_\phi(J_\phi + E\omega_\phi)}{H(\rho^2 + a_f^2)\sin^2\theta}, \qquad \dot{z} = HP_z - \frac{\omega_\psi(J_\psi - P_z\omega_\psi)}{H\rho^2\cos^2\theta}, \qquad \dot{\rho} = \frac{\rho^2 + a_f^2}{H(\rho^2 + a_f^2\cos^2\theta)}P_\rho, \\
&\dot{\theta} = \frac{1}{H(\rho^2 + a_f^2\cos^2\theta)}P_\theta, \qquad \dot{\phi} = \frac{J_\phi + E\omega_\phi}{H(\rho^2 + a_f^2)\sin^2\theta}, \qquad \dot{\psi} = \frac{J_\psi - P_z\omega_\psi}{H\rho^2\cos^2\theta} \\
&\dot{z}_{1,2} = \left(\frac{H_5}{H_1}\right)^{1/2}P_{1,2}, \qquad \dot{z}_{3,4} = \left(\frac{H_1}{H_5}\right)^{1/2}P_{3,4}.
\end{aligned} \tag{B2}$$

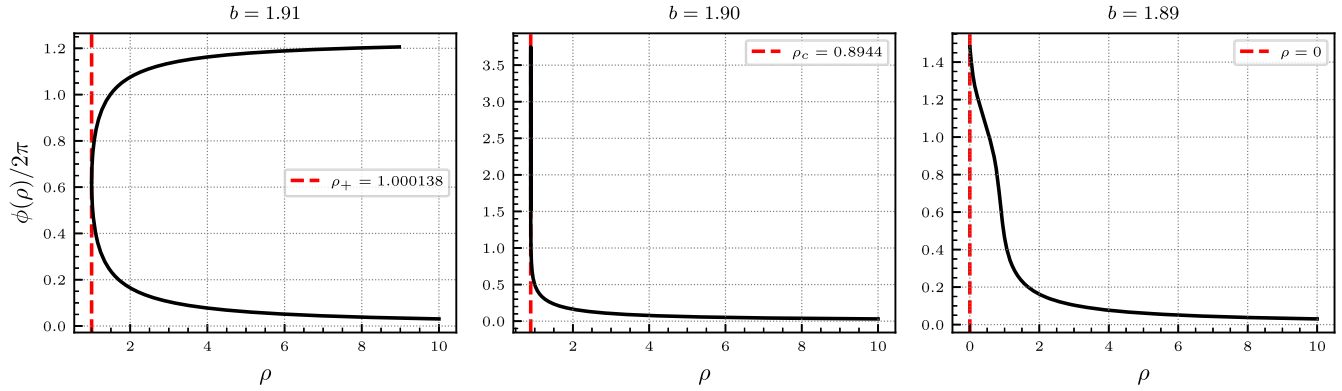

FIG. 35. Exact functions $\phi(\rho)/2\pi$ for the planar geometry. Left panel: overcritical case with $b = 1.91$. The vertical dashed line represents $\rho_+ = 1.000138$, which is the largest root of the radicand in Eq. (3.17). Central panel: critical case with $b = b_c = 1.9$. The vertical dashed line is the critical radius $\rho_c = 0.8944$, obtained from Eq. (3.19). Right panel: subcritical case with $b = 1.89$. The vertical dashed line is drawn for $\rho = 0$, below which the motion has no physical sense.

In order to find Eq. (3.17), we have to divide $\dot{\rho}$ by $\dot{t}$ and $\dot{\phi}$ by $\dot{t}$, where ω_ϕ and H are given in Eq. (3.7), $J_\phi = bE$ [from Eq. (3.16)], and P_ρ^2 is written in Eq. (3.15). By performing all these substitutions, the Eq. (3.17) is obtained.

APPENDIX C: DERIVATION OF THE GROUND TRUTH FOR THE PLANAR CASE

As hinted in Sec. III D, the planar case can be studied by exactly integrating the equatorial geodesics. The starting point is Eq. (3.17), from which we get

$$\phi(\rho) - \phi_0 = \pm \int_{\rho_0}^{\rho} \mathrm{d}\rho' \frac{(L_1 L_5 a_f + b\rho'^2)}{(\rho'^2 + a_f^2)\sqrt{\rho'^4 + \rho'^2(a_f^2 - b^2 + L_1^2 + L_5^2) - 2bL_1L_5a_f + (L_1^2 + L_5^2)a_f^2 + L_1^2L_5^2}}, \tag{C1}$$

which holds for all three of the regimes listed in Sec. III D (subcritical, critical, and overcritical). By numerically solving this integral, we can find the angle ϕ as a function of ρ and, by using Eq. (4.5), we can build the trajectory that we promote to ground truth in the comparison with the results obtained from the NN and the other numerical integrators. We have used extended-precision arithmetic to compute exactly the integral up to 32 digits. In the main text, we have set the geometry parameters to $a_f = 0.1$, $L_1 = L_5 = L = 1$, and $E = 1$. The functions $\phi(\rho)$, obtained by integrating Eq. (C1), are shown in Fig. 35 for $b = 1.91$ (overcritical regime), $b = b_c = 1.9$ (critical regime), and $b = 1.89$ (subcritical regime).

In the overcritical regime, the function $\phi(\rho)$ can be written in a closed way, which we now show for completeness. Indeed, when $b > b_c$ the shape of the potential $-Q_R$ is similar to the one in the bottom panel of Fig. 4, but shifted upwards and then intersecting the ρ axis twice. This implies that the radicand inside the integral (C1), proportional to Q_R^2, admits two distinct zeros. In terms of the variable y, defined in such a way that $\rho = a_f\sqrt{y}$, the integral becomes

$$\phi - \phi_0 = \frac{1}{2a_f^2} \int_{\rho_0^2/a_f^2}^{\rho^2/a_f^2} \mathrm{d}y \frac{(L_1L_5 + a_f b y)}{(1+y)\sqrt{y(y-y_+)(y-y_-)}}, \tag{C2}$$

where $y_\pm = \rho_\pm^2/a_f^2$. In particular,

$$y_\pm = \frac{b^2 - a_f^2 - L_1^2 - L_5^2 \pm \sqrt{(a_f + b - L_1 - L_5)(-a_f + b + L_1 - L_5)(-a_f + b - L_1 + L_5)(a_f + b + L_1 + L_5)}}{2a_f^2}. \tag{C3}$$

At the end of the day, Eq. (C2) can be written as

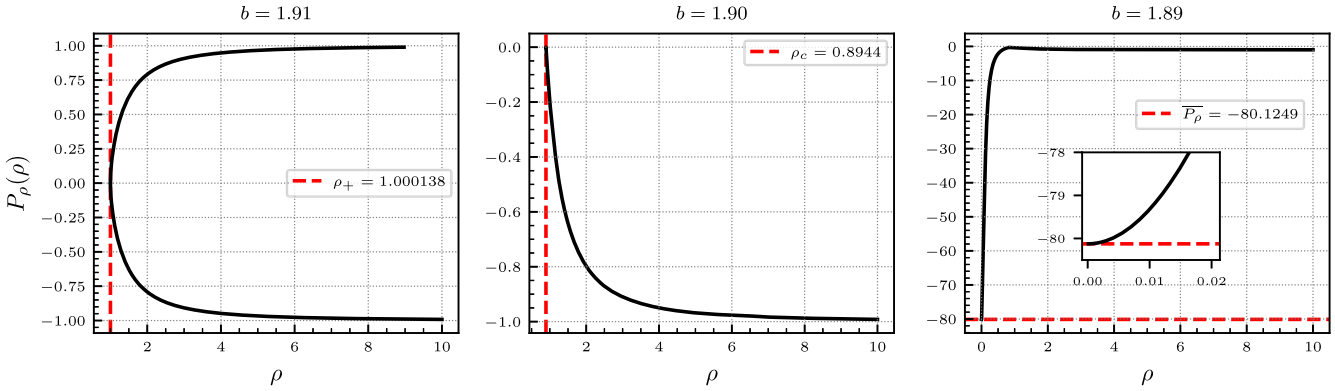

FIG. 36. Exact functions $P_\rho(\rho)$ for the planar geometry. Left panel: overcritical case with $b = 1.91$. The vertical dashed line represents $\rho_+ = 1.000138$, which is the largest root of the radicand in Eq. (3.17). Central panel: critical case with $b = b_c = 1.9$. The vertical dashed line is the critical radius $\rho_c = 0.8944$, obtained from Eq. (3.19). Right panel: subcritical case with $b = 1.89$. The horizontal dashed line represents $\overline{P_\rho} \equiv P_\rho(0) = -80.1249$, computed from Eq. (C7). The small frame is a zoomed-in view of the region $\rho \in [0, 0.02]$ and shows that $P_\rho(\rho)$ has a vanishing derivative for $\rho = 0$.

$$\phi - \phi_0 = \frac{1}{\rho_- a_f}\left[L_1 L_5\left(\Pi\left(-\frac{a_f^2}{\rho_-^2};\sin^{-1}\left(\frac{\rho_-}{\rho}\right)\middle|\frac{\rho_+^2}{\rho_-^2}\right) - \Pi\left(-\frac{a_f^2}{\rho_-^2};\sin^{-1}\left(\frac{\rho_-}{\rho_0}\right)\middle|\frac{\rho_+^2}{\rho_-^2}\right) - F\left(\sin^{-1}\left(\frac{\rho_-}{\rho}\right)\middle|\frac{\rho_+^2}{\rho_-^2}\right)\right.\right.$$
$$\left.\left. + F\left(\sin^{-1}\left(\frac{\rho_-}{\rho_0}\right)\middle|\frac{\rho_+^2}{\rho_-^2}\right)\right) - b a_f\left(\Pi\left(-\frac{a_f^2}{\rho_-^2};\sin^{-1}\left(\frac{\rho_-}{\rho}\right)\middle|\frac{\rho_+^2}{\rho_-^2}\right) - \Pi\left(-\frac{a_f^2}{\rho_-^2};\sin^{-1}\left(\frac{\rho_-}{\rho_0}\right)\middle|\frac{\rho_+^2}{\rho_-^2}\right)\right)\right], \tag{C4}$$

where $\Pi(n, \phi|m)$ is the incomplete elliptic integral of the third kind, and $F(\phi|m)$ is the incomplete elliptic integral of the first kind.

The exact function $P_\rho(\rho)$ is easily obtained by solving the constraint $\mathcal{H}(\rho) = 0$, with $\mathcal{H}$ being the Hamiltonian given in Eq. (3.13) at a fixed choice of the parameters. For their values reported at the beginning of this section, we find

$$b = 1.91 P_\rho(\rho) = \pm\sqrt{\frac{6380 - 16381\rho^2 + 10000\rho^4}{1 + 200\rho^2 + 10000\rho^4}}, \tag{C5}$$

$$b = 1.90 P_\rho(\rho) = -\frac{20(-4 + 5\rho^2)}{1 + 100\rho^2}, \tag{C6}$$

$$b = 1.89 P_\rho(\rho) = -\sqrt{\frac{6420 - 15621\rho^2 + 10000\rho^4}{1 + 200\rho^2 + 10000\rho^4}}. \tag{C7}$$

In the first equation, the minus sign holds for the incoming phase, while the plus sign is valid for the outgoing phase. The three functions are plotted in Fig. 36.

APPENDIX D: EXAMINATION OF THE GEOMETRY OF NONPLANAR CRITICAL GEODESICS

In this appendix, we investigate in more detail what has been said in Sec. III E for the nature of the nonplanar critical geodesics, showing that in general, they wrap around a spheroidal zone.

We solved the critical conditions $\mathcal{R}(\rho_c) = \mathcal{R}'(\rho_c) = 0$ in terms of ζ_c and b_c, getting Eq. (3.28). In order to find the critical radius ρ_c, we must work on the condition $P_\theta^2 \geq 0$. This can written as

$$\beta^2 - \frac{\beta_\phi^2}{\sin^2\theta} - \sin^2\theta \geq 0, \quad \text{with} \quad \beta = \frac{b}{a_f}, \qquad \beta_\phi = \frac{b_\phi}{a_f}. \tag{D1}$$

If we introduce $\sin^2\theta = \frac{1}{2}(1 - X)$, where $X = \cos(2\theta)$, the solutions of the associated equation are

$$X_\pm = 1 - \beta^2 \pm \sqrt{\beta^4 - 4\beta_\phi^2}. \tag{D2}$$

It is very easy to show that the discriminant $\beta^4 - 4\beta_\phi^2$ is always positive in the critical regime (3.28). From the definition of X, we can write down the following chain of inequalities:

$$-1 \leq X_- \leq X_+ \leq 1. \tag{D3}$$

The condition $X_- \geq -1$ is always false, while $X_+ \leq 1$ is always verified, so that the first nontrivial condition is $X_+ \geq -1$, which yields

$$\beta^2 - 2 - \sqrt{\beta^4 - 4\beta_\phi^2} \leq 0 \leftrightarrow \sqrt{b^4 - 4b_\phi^2 a_f^2} \geq b^2 - 2a_f^2. \tag{D4}$$

Since we want to study the critical geodesics, we should evaluate this relation in the critical regime (3.28). In doing so, we have to distinguish two cases depending on the sign of the right-hand side. It turns out that if $b_c^2 < 2a_f^2$, then from Eq. (3.28), $2\rho_c^2 + L_1^2 + L_5^2 < 0$, and it is never satisfied, so the only case to consider is when $b_c^2 > 2a_f^2$. This brings us to the following condition:

$$b_c^2 \geq b_{\phi,c}^2 + a_f^2. \tag{D5}$$

When we substitute the expressions in Eq. (3.28), we have to distinguish two cases according to the sign of ζ_c, with the aim of solving with respect to ρ_c:

$$\begin{aligned} a_f\zeta_c = \rho_c^2 + a_f^2 &\Longrightarrow 0 \leq \rho_c \leq \sqrt{a_f|L_1 - L_5| - L_1L_5} \quad \text{for } a_f \geq \frac{L_1L_5}{|L_1 - L_5|} \\ a_f\zeta_c = -(\rho_c^2 + a_f^2) &\Longrightarrow \begin{cases} \sqrt{L_1L_5 - a_f(L_1 + L_5)} \leq \rho_c \leq \sqrt{L_1L_5 + a_f(L_1 + L_5)}, & 0 < a_f \leq \frac{L_1L_5}{L_1+L_5} \\ 0 \leq \rho_c \leq \sqrt{a_f(L_1 + L_5) - L_1L_5}, & a_f > \frac{L_1L_5}{L_1+L_5}. \end{cases} \end{aligned} \tag{D6}$$

The former case corresponds to a *corotating* geodesic, and the latter to a *counterrotating* geodesic.

The inequality (D1) has as its solution $X_- \leq X \leq X_+$. Since $X_- < -1$ is always valid, the first inequality is always satisfied. Hence, it is left to solve $\cos(2\theta) \leq X_+$, whose solution is (keeping in mind that $\theta \in [0, \pi]$)

$$\begin{aligned} &\theta_{\text{bound}} \leq \theta \leq \pi - \theta_{\text{bound}}, \\ &\text{with} \quad \theta_{\text{bound}} = \frac{\arccos(X_+)}{2}. \end{aligned} \tag{D7}$$

Since ϕ varies from 0 to 2π, the geodesic describes a spheroidal zone. In order to understand what it is, we recall that the change of coordinates in Eq. (3.22) is a parametrization of an ellipsoid, at ρ fixed, whose Cartesian equation is

$$\frac{x^2}{\rho^2 + a_f^2} + \frac{y^2}{\rho^2 + a_f^2} + \frac{z^2}{\rho^2} = 1. \tag{D8}$$

By indicating in this context $a^2 = \rho^2 + a_f^2$, $c^2 = \rho^2 + a_f^2$, $d^2 = \rho^2$, then, if two of these quantities are equal, the ellipsoid is said to be a spheroid (or ellipsoid of revolution), and since $a = c > d$, then it is an oblate spheroid. As θ varies inside Eq. (D7), the surface is the one bounded by two planes that are parallel to the x-y plane and intersect the oblate spheroid at $\theta = \theta_{\text{bound}}$ and $\theta = \pi - \theta_{\text{bound}}$. This region is called the spheroid zone.

In the text, we considered the case with $X_+ = 1$, corresponding to $\theta_{\text{bound}} = 0$, and hence θ varies within its interval of definition, $\theta \in [0, \pi]$. For this reason, such trajectory wraps around the complete oblate spheroid in Eq. (D8) with $\rho = \rho_c$. Furthermore, the condition $X_+ = 1$ brings to $\beta_{\phi,c} = 0 \leftrightarrow b_{\phi,c} = 0$, and so from the definition (3.28), we gain $\zeta_c = -\frac{L_1L_5}{a_f}$. This explains the first condition of Eq. (3.30), and hence, the fact that this geodesic is counterrotating. The critical radius can be found from $X_+ = 1$ expressed in terms of ρ_c and then solving it with respect to ρ_c, or also from the first equation of (3.28) by substituting ζ_c^2 with $(L_1L_5/a_f)^2$ and solving with respect to ρ_c (which is, by definition, non-negative). At the end, we get

$$\rho_c = \sqrt{L_1L_5 - a_f^2}, \qquad 0 < a_f \leq \sqrt{L_1L_5}, \tag{D9}$$

which is exactly (3.29). Notice that this condition has to satisfy Eq. (D6). It turns out that $\sqrt{L1L5} > \frac{L1L5}{L1+L5}$ for every $L_1, L_5 > 0$, and it can be shown that if a_f satisfies the first or second condition of (D6), then the corresponding critical radius ρ_c falls within the associated interval.

APPENDIX E: RESULTS OF THE NN EXPLOITING SEPARABILITY OF THE MOTION

In this appendix, we show the results obtained using the NN by integrating the equations coming from separability —i.e., Eq. (3.17). This exercise goes beyond the scope of this paper, and therefore, we will keep the discussion of the results short. We suggest to read first Sec. IV, where we have introduced most of the ingredients that will be reused in this appendix. We do not present a comparison with other integration methods, nor the errors committed by the NN, but we compare the predicted geodesic in the Cartesian plane with the ground truth, and we highlight the main differences with the results and the strategies presented in the main text.

In this case, we cannot speak about a Hamiltonian neural network, since the equations to solve are not the Hamilton equations, and the energy conservation cannot be imposed

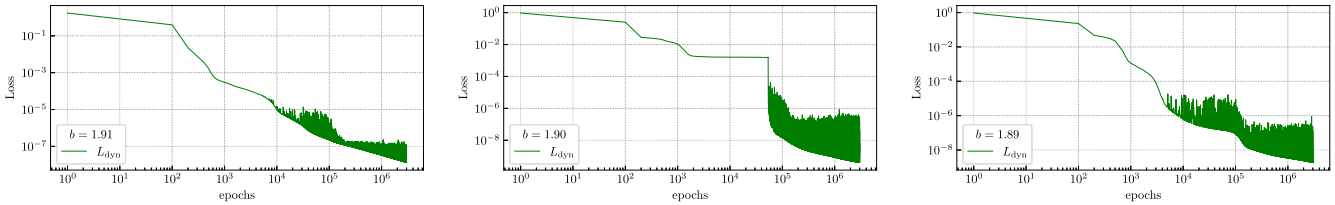

FIG. 37. Dynamical loss functions corresponding to the equations of motion obtained by exploiting the separability of the geometry. From left to right: overcritical case with $b = 1.91$, critical case with $b = b_c = 1.9$, and subcritical case with $b = 1.89$. In all the cases, the number of points is $N = 800$.

at the level of the loss function. Still, the strategy presented in Sec. II applies to any set of differential equations, regardless of whether they are Hamilton or different ones. We will label the results predicted by the neural network with "NN."

In this case, the NN maps the input time[12] t to an output layer with two nodes, which are denoted by O_ρ and O_ϕ. The solution to the system is constructed as

$$\hat{\rho}(t, \mathbf{w}) = \rho_0 + f(t) O_\rho(t, \mathbf{w}), \tag{E1}$$

$$\hat{\phi}(t, \mathbf{w}) = \phi_0 + f(t) O_\phi(t, \mathbf{w}). \tag{E2}$$

The weights $\mathbf{w}$ of the NN are found by minimizing only the loss function $L_{\rm dyn}$, since $L_{\rm energy}$ in this case is not defined. Therefore, there is no need to tune the parameter λ.

We fix the parameters of the geometry to those already used in Sec. IV, and we treat the three regimes: overcritical with $b = 1.91$, critical with $b = b_c = 1.9$, and subcritical with $b = 1.89$. ρ_0 and ϕ_0 remain the same, specified in Sec. IV for the three situations. For all the cases, we have employed a neural network consisting of two hidden layers with 64 neurons each. The training is performed over 3×10^6 epochs by varying the learning rate, as done in Sec. IV; furthermore, the training interval is $\mathcal{T}_{\rm train}$ (different for each b), and the absence of overfitting is explicitly checked by validating the loss function on $\mathcal{T}_{\rm val}$ at the end of the training.

We move now to present the results individually for the three cases.

1. Overcritical case

We set $b = 1.91$ and train the neural network over the interval $\mathcal{T}_{\rm train} = [0, 15.3157]$ divided into $N = 800$ points. The loss function is shown in the left panel of Fig. 37. The reason behind such a particular choice for the upper bound is that the particle motion has an incoming and an outgoing phase. The former is given by considering the negative sign in Eq. (3.17) and corresponds to a decrease from ρ_0 to the minimum value $\rho_+ = 1.000138$, where $d\rho/dt|_{\rho=\rho_+} = 0$. The latter is given by taking the positive sign, and it corresponds to an increase from ρ_+ to ∞. We obtain the outgoing solution just from the reflection of the incoming one, since the system is symmetric under $\rho \mapsto -\rho$. In order to do so, we need the time t_+ such that $\rho(t_+) = \rho_+$, and we obtain the value $t_+ = 15.3157$ from the `RK45` method. Notice that for $t > t_+$, the radicand in Eq. (3.17) becomes negative; we are therefore forced to stop computing the prediction at this time. In the Hamilton case presented in Sec. IV A, we do not need to choose carefully $\mathcal{T}_{\rm train}$, since the information about the incoming and outgoing motion is automatically incorporated into the Hamilton equations.

The predicted $\rho(t)$ and $\phi(t)/2\pi$ are shown in the left panel of Fig. 38 for both the incoming and outgoing phases. The value t_+ is highlighted by a vertical black line, and we find $\hat{\rho}(t_+) = 1.00020$, which is at 0.006% away from ρ_+. The behavior in the coordinate time t of the two variables is completely analogous to that with respect to the affine parameter s obtained in Fig. 8.

The geodesic in the Cartesian plane is obtained by using Eq. (4.5) and is compared to the ground truth in the left panel of Fig. 39. The predicted trajectory reproduces perfectly the ground truth.

2. Critical case

In order to obtain the critical regime, we set $b = b_c = 1.9$. In this case, the coordinate ρ of the particle has to decrease from ρ_0 to reach the critical radius $\rho_c = 0.8944$, and then it has to remain on this value. Since there is no growing, we only consider the equation with the negative sign. In this case, the radicand in Eq. (3.17) has only one root, at ρ_c, and then it can be written as $(\rho^2 - \rho_c^2)^2$. Therefore, it presents no problems, and the solution can be computed at arbitrarily large times. We have trained the neural network in the interval $\mathcal{T}_{\rm train} = [0, 140]$, and the loss function is shown in the central panel of Fig. 37.

[12]Notice that now the equations of motion describe the dynamics in terms of the coordinate time t. Nonetheless, the trajectories in the Cartesian plane have to be the same as those that would be obtained by solving the Hamilton equations with respect to the affine parameter s. Of course, the distribution of the points along the geodesics is different, but the final result is the same if we take the number of such points much larger than 1 (as we have always done).

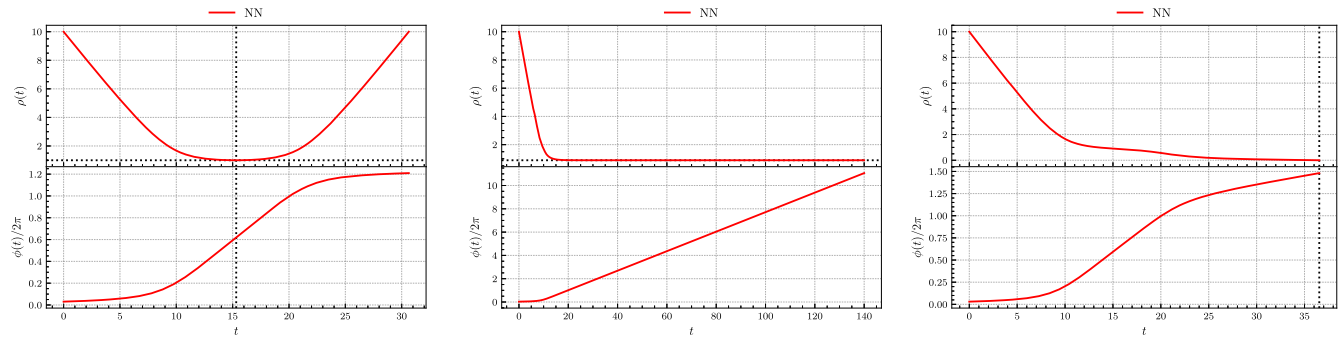

FIG. 38. Predicted $\rho(t)$ (top panels) and $\phi(t)/2\pi$ (bottom panels) corresponding to the equations of motion obtained by exploiting the separability of the geometry. Left panel: overcritical case with $b = 1.91$. The vertical dashed line corresponds to $t_+ = 15.3157$. The functions for $t > t_+$ (outgoing motion) are obtained from the reflection of the incoming phase. The horizontal dashed line in the top panel represents $\rho_+ = 1.000138$. Central panel: critical case with $b = b_c = 1.9$. The horizontal dashed line in the top panel represents $\rho_c = 0.8944$. Right panel: subcritical case with $b = 1.89$. The vertical dashed line corresponds to the collision time $\bar{t} = 36.5417$.

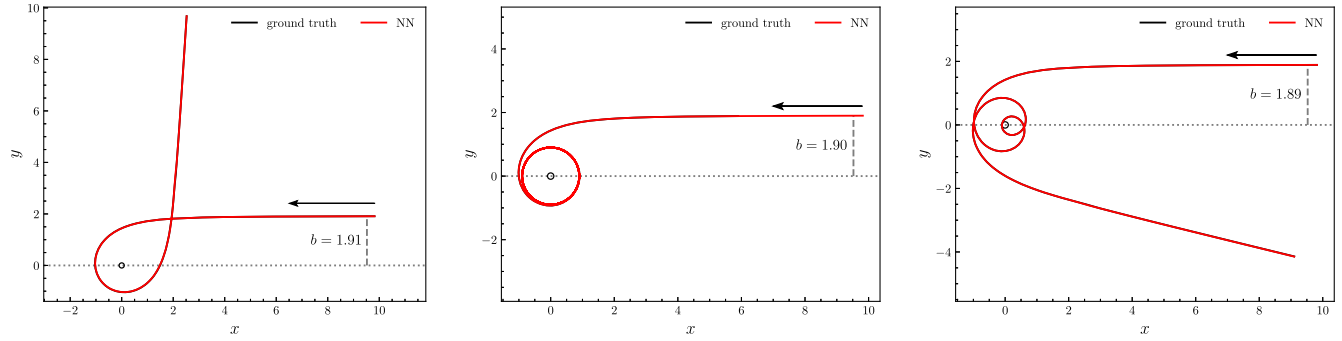

FIG. 39. Comparison between the geodesic predicted by the NN and the ground truth in the Cartesian plane. From left to right: overcritical case with $b = 1.91$, critical case with $b = b_c = 1.9$, and subcritical case with $b = 1.89$. The small black circle has radius $a_f = 0.1$, and it represents the fuzzball.

The predicted functions $\rho(t)$ and $\phi(t)$ are shown in the central panel of Fig. 38. The value ρ_c is represented by the black line in the top panel and, as we can see, $\rho(t)$ remains stable on it. Like the Hamilton case, the NN is confirmed to be able to provide stable and accurate predictions for trajectories over a long timescale even without the need to impose the conservation of the energy.

The resulting geodesic in the Cartesian plane is shown in the central panel of Fig. 39, and it accurately reproduces the ground truth.

3. Subcritical case

In order to obtain the subcritical regime, we set $b = 1.89$, and we train the neural network in the interval $\mathcal{T}_{\text{train}} = [0, 37]$ divided into $N = 800$ points. The loss function is shown in the right panel of Fig. 37.

In Sec. IV C, we pointed out the difficulty in solving the Hamilton equations due to the steep decrease of $P_\rho(s)$ approaching the time $\bar{s}$, and we proposed two different strategies. In this case, there is no dynamical equation for P_ρ thanks to the separability, and such difficulty does not arise. In addition, the radicand in Eq. (3.17) is always positive definite.

The particle trajectory still describes an incoming and outgoing motion, and the transition is determined by the minimum value $\rho = 0$ that occurs at a given time $\bar{t}$. We follow the same strategy as Sec. IV C 1 of determining $\bar{t}$ by the interpolation of $\rho(t)$ with 0 (recall Fig. 19), and we obtain the outgoing phase by reflection of the incoming one. The functions $\rho(t)$ and $\phi(t)/2\pi$ are shown in the right-panel of Fig. 38, and from the aforementioned procedure, we find $\bar{t} = 36.5417$, which is marked by the vertical black line.

Finally, the full geodesic projected onto the Cartesian plane is shown in the right panel of Fig. 39, and it is in perfect agreement with the ground truth.

[1] Sam Greydanus, Misko Dzamba, and Jason Yosinski, Hamiltonian neural networks, arXiv:1906.01563.

[2] Marios Mattheakis, David Sondak, Akshunna S. Dogra, and Pavlos Protopapas, Hamiltonian neural networks for solving equations of motion, Phys. Rev. E **105**, 065305 (2022).

[3] Miles Cranmer, Sam Greydanus, Stephan Hoyer, Peter Battaglia, David Spergel, and Shirley Ho, Lagrangian neural networks, arXiv:2003.04630.

[4] Chen-Di Han, Bryan Glaz, Mulugeta Haile, and Ying-Cheng Lai, Adaptable Hamiltonian neural networks, Phys. Rev. Res. **3**, 023156 (2021).

[5] Sam Greydanus, Misko Dzamba, and Jason Yosinski, Hamiltonian neural networks, arXiv:1906.01563.

[6] Teo Deveney, Eike Mueller, and Tony Shardlow, A deep surrogate approach to efficient Bayesian inversion in PDE and integral equation models (2021).

[7] Maziar Raissi, Paris Perdikaris, and George Em Karniadakis, Physics informed deep learning (part I): Data-driven solutions of nonlinear partial differential equations, J. Comput. Phys. **378**, 686 (2019).

[8] Benjamin Wu, Oliver Hennigh, Jan Kautz, Sanjay Choudhry, and Wonmin Byeon, Physics informed RNN-DCT networks for time-dependent partial differential equations, arXiv:2202.12358.

[9] Alan S. Cornell, Anele Ncube, and Gerhard Harmsen, Using physics-informed neural networks to compute quasinormal modes, Phys. Rev. D **106**, 124047 (2022).

[10] Aarjav Jain, Challenger Mishra, and Pietro Liò, A physics-informed search for metric solutions to Ricci flow, their embeddings, and visualisation (2022).

[11] Raimon Luna, Juan Calderón Bustillo, Juan José Seoane Martínez, Alejandro Torres-Forné, and José A. Font, Solving the Teukolsky equation with physics-informed neural networks, Phys. Rev. D **107**, 064025 (2023).

[12] Raimon Luna, Daniela D. Doneva, José A. Font, Jr-Hua Lien, and Stoytcho S. Yazadjiev, Quasinormal modes in modified gravity using physics-informed neural networks, Phys. Rev. D **109**, 124064 (2024).

[13] Ru Geng, Yixian Gao, Jian Zu, and Hong-Kun Zhang, Graph attention Hamiltonian neural networks: A lattice system analysis model based on structural learning (2024).

[14] Gilles Louppe, Kyunghyun Cho, Cyril Becot, and Kyle Cranmer, QCD-aware recursive neural networks for jet physics, J. High Energy Phys. 01 (2019) 057.

[15] Mohammad Amin Nabian and Hadi Meidani, Adaptive physics-informed neural networks for Markov-chain Monte Carlo, arXiv:2008.01604.

[16] B. Hashemi, R. G. Corominas, and A. Giacchetto, Can transformers do enumerative geometry?, arXiv:2408.14915.

[17] I. E. Lagaris, A. Likas, and D. I. Fotiadis, Can transformers do enumerative geometry?, IEEE Trans. Neural Networks **9**, **987** (1998).

[18] Alex Cole, Andreas Schachner, and Gary Shiu, Searching the landscape of flux vacua with genetic algorithms, J. High Energy Phys. 11 (2019) 045.

[19] Alex Cole, Sven Krippendorf, Andreas Schachner, and Gary Shiu, Probing the structure of string theory vacua with genetic algorithms and reinforcement learning, in *35th Conference on Neural Information Processing Systems* (NeurIPS, 2021).

[20] Nate MacFadden, Andreas Schachner, and Elijah Sheridan, The DNA of Calabi-Yau hypersurfaces (2024).

[21] Stefano Lanza, Machine learning the breakdown of tame effective theories, Eur. Phys. J. C **84**, 631 (2024).

[22] Stefano Lanza, Neural network learning and quantum gravity, J. High Energy Phys. 07 (2024) 105.

[23] Rehan Deen, Yang-Hui He, Seung-Joo Lee, and Andre Lukas, Machine learning string standard models, Phys. Rev. D **105**, 046001 (2022).

[24] Per Berglund, Giorgi Butbaia, Tristan Hübsch, Vishnu Jejjala, Damián Mayorga Peña, Challenger Mishra, and Justin Tan, Precision string phenomenology, Phys. Rev. D **111**, 086007 (2025).

[25] Anthony Ashmore, Yang-Hui He, and Burt A. Ovrut, Machine learning Calabi–Yau metrics, Fortschr. Phys. **68**, 2000068 (2020).

[26] Vishnu Jejjala, Damian Kaloni Mayorga Pena, and Challenger Mishra, Neural network approximations for Calabi-Yau metrics, J. High Energy Phys. 08 (2022) 105.

[27] Magdalena Larfors, Andre Lukas, Fabian Ruehle, and Robin Schneider, Learning size and shape of Calabi-Yau spaces (2021).

[28] Per Berglund, Giorgi Butbaia, Tristan Hüubsch, Vishnu Jejjala, Damián Mayorga Peña, Challenger Mishra, and Justin Tan, Machine-learned Calabi–Yau metrics and curvature, Adv. Theor. Math. Phys. **27**, 1107 (2023).

[29] Per Berglund, Yang-Hui He, Elli Heyes, Edward Hirst, Vishnu Jejjala, and Andre Lukas, New Calabi–Yau manifolds from genetic algorithms, Phys. Lett. B **850**, 138504 (2024).

[30] Lara B. Anderson, James Gray, and Magdalena Larfors, Lectures on numerical and machine learning methods for approximating Ricci-flat Calabi-Yau metrics (2023).

[31] Yacoub Hendi, Magdalena Larfors, and Moritz Walden, Learning group invariant Calabi-Yau metrics by fundamental domain projections, Mach. Learn. Sci. Tech. **6**, 015050 (2025).

[32] Viktor Mirjanić and Challenger Mishra, Symbolic approximations to Ricci-flat metrics via extrinsic symmetries of Calabi-Yau hypersurfaces (2024).

[33] Rak-Kyeong Seong, Unsupervised machine learning techniques for exploring tropical coamoeba, brane tilings and Seiberg duality, Phys. Rev. D **108**, 106009 (2023).

[34] Eugene Choi and Rak-Kyeong Seong, Machine learning regularization for the minimum volume formula of toric Calabi-Yau 3-folds, Phys. Rev. D **109**, 046015 (2024).

[35] Rak-Kyeong Seong, Generative AI for brane configurations and coamoeba, Phys. Rev. D **111**, 086013 (2025).

[36] Yang-Hui He, Vishnu Jejjala, and Tomás S. R. Silva, Metaheuristic generation of brane tilings, J. High Energy Phys. 04 (2024) 149.

[37] M. Bianchi, D. Consoli, and J. F. Morales, Probing fuzzballs with particles, waves and strings, J. High Energy Phys. 06 (2018) 157.

[38] Massimo Bianchi and Giorgio Di Russo, 2-charge circular fuzz-balls and their perturbations, J. High Energy Phys. 08 (2023) 217.

[39] Giorgio Di Russo, Francesco Fucito, and Jose Francisco Morales, Tidal resonances for fuzzballs (2024).

[40] Massimo Bianchi, Carlo Di Benedetto, Giorgio Di Russo, and Giuseppe Sudano, Charge instability of JMaRT geometries, J. High Energy Phys. 09 (2023) 078.
[41] Massimo Bianchi, Giorgio Di Russo, Alfredo Grillo, Jose Francisco Morales, and Giuseppe Sudano, On the stability and deformability of top stars, J. High Energy Phys. 12 (2023) 121.
[42] Andrea Cipriani, Carlo Di Benedetto, Giorgio Di Russo, Alfredo Grillo, and Giuseppe Sudano, Charge (in)stability and superradiance of topological stars, J. High Energy Phys. 07 (2024) 143.
[43] K. Hornik, M. Stinchcombe, and H. White, Multilayer feedforward networks are universal approximators, Neural Netw. **2**, **359** (1989).
[44] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning* (MIT Press, 2016).
[45] Samir D. Mathur, The information paradox: A pedagogical introduction, Classical Quantum Gravity **26**, 224001 (2009).
[46] Jacob D. Bekenstein, Extraction of energy and charge from a black hole, Phys. Rev. D **7**, 949 (1973).
[47] Oleg Lunin and Samir D. Mathur, Metric of the multiply wound rotating string, Nucl. Phys. **B610**, 49 (2001).
[48] Samir D. Mathur, The fuzzball proposal for black holes: An elementary review, Fortschr. Phys. **53**, 793 (2005).
[49] Iosif Bena, Emil J. Martinec, Samir D. Mathur, and Nicholas P. Warner, Fuzzballs and microstate geometries: Black-hole structure in string theory (2022).
[50] Yuri Chervonyi and Oleg Lunin, (Non)-integrability of geodesics in D-brane backgrounds, J. High Energy Phys. 02 (2014) 061.
[51] S. Chandrasekhar, The mathematical theory of black holes, Fundam. Theor. Phys. **9**, 5 (1984).
[52] M. Bianchi, D. Consoli, A. Grillo, and J. F. Morales, Light rings of five-dimensional geometries, J. High Energy Phys. 03 (2021) 210.
[53] Massimo Bianchi and Giorgio Di Russo, Turning black holes and D-branes inside out of their photon spheres, Phys. Rev. D **105**, 126007 (2022).
[54] Massimo Bianchi and Giorgio Di Russo, Turning rotating D-branes and black holes inside out their photon-halo, Phys. Rev. D **106**, 086009 (2022).
[55] https://gitlab.com/AleDS/hnn_fuzzball_geodesics.
[56] François Chollet *et al.*, Keras, https://keras.io (2015).
[57] Martín Abadi *et al.*, TensorFlow: Large-scale machine learning on heterogeneous systems, Software available from https://www.tensorflow.org/ (2015).
[58] Diederik P. Kingma and Jimmy Ba, Adam: A method for stochastic optimization (2017).
[59] `solve_ivp` by SciPy, https://docs.scipy.org/doc/scipy/reference/generated/scipy.integrate.solve_ivp.html.
[60] M. Bianchi, Dario Consoli, Alfredo Grillo, and Jose F. Morales, The dark side of fuzzball geometries, J. High Energy Phys. 05 (2019) 126.