

Machine learning methods for probabilistic locked-mode predictors in tokamak plasmas

Cihan Akçay,¹ John M. Finn,¹ Dylan P. Brennan,² Thomas Burr,³ and Doğa M. Kürkçüoğlu⁴

¹*Tibbar Plasma Technologies, Incorporated, 274 DP Rd., Los Alamos, NM 87544*^{a)}

²*Princeton University, Princeton, NJ 08544*

³*Los Alamos National Laboratory, NM 87544*

⁴*Fermilab*

A rotating tokamak plasma can interact resonantly with external helical magnetic perturbations, also known as error fields. This can lead to locking, which can lead to disruptions. We leverage machine learning (ML) methods to train them to predict locking events. We use a simple coupled third order nonlinear ODE model to represent the interaction of the magnetic perturbation and the plasma rotation with the error field. This model is sufficient to describe qualitatively the locking and unlocking bifurcations. We explore using ML algorithms with simulation data and experimental data, focusing on methods that can be used with such necessarily sparse data sets. These methods lead to the possibility of the avoidance of locking in real time operations. We also discuss an analogy with the Van der Waals (VDW) equation of state. We describe the operational space in terms of two control parameters: the magnitude of the error field and the rotation frequency associated with the momentum source that maintains the plasma rotation. The outcomes are quantified by *order parameters*, i.e. parameters that completely characterize the state, whether locked or unlocked. We use unsupervised ML methods to classify locked and unlocked states, and note the usefulness of a certain normalization of the order parameters. We then supervised ML methods to estimate the probability of locking in the region of control parameter space with hysteresis, i.e. the set of control parameters for which both locked and unlocked states can exist. We test three ML methods and show that a neural network gives the best estimate of this locking probability.

I. INTRODUCTION

The MHD response of a rotating toroidal plasma to non-axisymmetric error fields^{1,2}, a form of driven magnetic reconnection or driven tearing mode activity, can be important in tokamaks because it can cause disruptions. A disruption in a tokamak, especially a major one, involves a rapid loss of plasma confinement. Specifically, during a major disruption the rapid loss of confinement can lead to a fast release of thermal energy. Such disruptions can cause both surface melting of plasma-facing components and high electromagnetic loads.

Error fields can arise due to imperfections or alignment errors in external coils, or due to disadvantageously placed current feeds. Error fields can cause locking of a rotating plasma by exerting a large Maxwell torque², which leads to amplification of the driven magnetic perturbations, with the magnetic perturbation and the decreased rotation synergistically amplifying each other. In many tokamak experiments momentum is injected, e.g. by unbalanced neutral beams, to maintain the rotation to prevent locking, but this mechanism will be unavailable in ITER^{3,4}. Disruption avoidance, by means other than injecting momentum, requires understanding and forecasting the response of a plasma to error fields in the presence of plasma rotation.

Disruption forecasting/avoidance via ML techniques have been common practice in tokamak operations for nearly two decades. Artificial neural networks were trained on the available diagnostic data as early as mid 2000's to detect disruptions⁵⁻⁷. More recently, a disruption predictor

based on deep learning, combining recurrent and convolutional neural networks, was developed by Kates-Harbeck et al.⁸. Fu et al.⁹ applied ensemble methods to DIII-D data to judge the 'tearability' or 'disruptivity' of their plasmas for real-time feedback control where their tearing mode predictor is used to control the neutral beam power in order to affect the plasma rotation. Because of the danger posed by disruptions, any such detection of ITER disruptions should require an extremely low false negative (where 'negative' means unlocked) rate, and achieving this is a serious challenge for ML algorithms.

This paper presents a proof-of-principle of using machine learning classifiers (MLC's) to forecast mode-locking by calculating the probability of a rotating plasma to lock to a static error field. The appeal of this approach is the possibility of generating a meaningful, quantitative result, i.e. the locking probability, using sparse data. This capability is relevant for realistic scenarios where the amounts of data that can be obtained from either experiments or high-fidelity simulations can be limited. The data sparsity is simulated here by having a single measurement (of several physical quantities) for each point in the parameter space and then going to coarser grids in the parameter space to determine where the accuracy of the MLC's breaks down.

The data required for the MLC's are generated by solving a third order system of ODE's that describes the locking dynamics. The advantage of this ODE model is the rapid generation of data that takes a few minutes on a few dozen processors, as well as the success of similar modeling^{1,2,10} in capturing the basic aspects of the locking dynamics that are useful for the MLC's.

Another advantage of the present method is its probabilistic interpretation of locking. The need for this probability originates from the solutions of the ODE model that exhibit

^{a)}Electronic mail: c_akcay@tibbartech.com

hysteresis and bifurcations between the low-rotation (locked) and high-rotation (unlocked) branches. This implies an inherent sensitivity to noise or perturbations in the initial conditions. In such hysteretic systems, knowing the probability of locking, conditional on the controls parameters of the system, can be insightful because it quantifies the sensitivity to sudden changes in the plasma conditions. In fact, this probability is a measure of the robustness of an unlocked state to a disturbance like a large sawtooth or a large edge-localized mode (ELM)¹¹. The capability described here, when trained or calibrated on real-time experimental data, has the potential to become a forecasting tool for disruptions, which can be used in active feedback control.

The third order ODE model has three dependent variables $\tilde{\psi}_t$, θ_t , and Ω_t , which represent the magnitude of the magnetic field perturbation, its phase relative to that of the error field, and the toroidal plasma rotation frequency, all at the mode rational surface. These, in the time-asymptotic state, are the *order parameters* of the system. The initial condition for each order parameter is sampled randomly over a prescribed range. The *control parameters* of the model are the magnitude ψ_w of the error field at the edge and the frequency Ω_0 of a momentum source driving the plasma rotation. We choose a uniform grid with a nominal resolution of 200×200 in this 2D control space. Other parameters in the model are kept fixed.

The equations advancing $\tilde{\psi}_t$ and θ_t represent the linear growth of a weakly stable (intrinsic) tearing mode which is driven by the error field $\tilde{\psi}_w$ in the presence of rotation Ω_t . Spontaneous stability is assured by setting the stability parameter of the mode Δ_1 , one of the fixed parameters of the model, to a small but negative number. For simplicity the tearing mode is taken to be in the viscoresistive (VR) regime¹². The equation for Ω_t represents the competition between the quasilinear Maxwell torque^{2,13} slowing the plasma down and the momentum source driving $\Omega_t \rightarrow \Omega_0$.

We examine two cases: one that is *biased* toward locking and one that is relatively unbiased or *neutral*. These two cases represent, respectively: a tokamak with a high level of fluctuations such as sawteeth or ELM's which may cause locking, and another more quiescent tokamak that operates in a safer regime. The chosen range in the initial conditions controls the amount of the said bias, i.e. pre-determines the tendency of the system to lock.

The ODE model has time-asymptotic solutions that are steady state, consisting of locked and unlocked states. Loosely speaking, the locked states are characterized as having large steady-state values of $\tilde{\psi}_t$, $\theta_t \approx 0$, and Ω_t small; unlocked states have small steady-state values of $\tilde{\psi}_t$, $\theta_t \approx -\pi/2$, and large values of Ω_t . Less ambiguous criteria for the locked and unlocked states of the plasma are determined by ML classifiers. These are aided greatly by normalizing the order parameters to span $[0, 1]$. Under this normalization the phase θ_t is redundant. The normalization of the remaining two order parameters onto a unit square via $(\tilde{\psi}_n, \Omega_n) \equiv (|\Delta_1| \tilde{\psi}_t / \tilde{\psi}_w, \Omega_t / \Omega_0)$ redistributes more than 99% of the solutions into two tight clusters, representing the locked and unlocked states of the plasma. The locked solutions cluster around $(\tilde{\psi}_n, \Omega_n) \approx (1, 0)$, while the unlocked

solutions cluster around $(\tilde{\psi}_n, \Omega_n) \approx (0, 1)$.

The steady state solutions of this model can be found analytically by solving a cubic equation in Ω_t , representing the torque balance on the plasma at the rational surface (see Refs. 1,2). These solutions show the presence of a *hysteretic* region in $(\tilde{\psi}_w, \Omega_0)$, where the cubic equation of torque balance has three real roots. Two of these 3 roots, the *attractors* of the system, represent the locked and unlocked states of the plasma. The hysteretic region is separated in control parameter space from two other regions, where the cubic equation has strictly one real root. In one of the regions, the states are locked and in the other unlocked. These two regions are separated from the hysteretic region by analytically defined boundary segments that we call the bifurcation boundary segments. These two segments come together at a critical point (CP) below which smooth transitions are seen to occur between the locked and unlocked states. This is analogous to the CP encountered in liquid-to-gas phase transitions, as described by the VDW equation of state.

For the ODE system presented here, the *hysteresis* described in the preceding paragraph appears as a sizable region in the 2D control space of $(\tilde{\psi}_w, \Omega_0)$, featuring a strong mixing of locked and unlocked solutions, often neighboring each other. This mixing implies a sensitivity of the ODE solutions in the hysteretic region to the randomly chosen initial conditions, whose range in fact influences the density of locked and unlocked states within the hysteretic region. This implies, for a different realization of the random initial conditions, that a point identified previously as locked can jump to an unlocked state. The probability of locking, conditional on the controls parameters of the system, can be insightful because it quantifies the system's sensitivity to noise or to sudden changes in the plasma conditions. In our model, this probability is a measure of the size of the domain of attraction of the locked root on the slow-rotation branch, given the domain of initial guesses. Or in the context of tokamak physics, the said probability is a measure of the robustness of an unlocked state to a disturbance like a large sawtooth or a big ELM.

For applying ML methods to the problem of locking probabilities, we follow a two-fold approach. The first stage aims to classify unambiguously each solution of the ODE system as locked or unlocked in the space of the *normalized* order parameters, without human input. This is accomplished by subjecting the input vector comprising all of the values of the normalized order parameters to *K-means clustering* (KMC), which is an unsupervised classification algorithm. KMC is a geometric method that makes a 'hard' assignment to each sample. The classification results have also been benchmarked with Gaussian mixture models, a probabilistic method that fits anisotropic Gaussians to the data.

The second stage involves training a series of supervised classifiers to calculate the conditional probability of locking $p(L|\tilde{\psi}_w, \Omega_0)$ in the control space $(\tilde{\psi}_w, \Omega_0)$. The locking probability satisfies $0 \lesssim p(L|\tilde{\psi}_w, \Omega_0) \lesssim 1$ within the hysteretic region, and an area where $p(L|\tilde{\psi}_w, \Omega_0)$ is close to one is an area susceptible to locking.

The first step in calculating $p(L|\tilde{\psi}_w, \Omega_0)$ entails switching from the 2D space of normalized order parameters $(\tilde{\psi}_n, \Omega_n)$

to the 2D space of the control parameters $(\tilde{\psi}_w, \Omega_0)$. Next, a pre-processing step is carried out where the binary class labels (unlocked and locked) emerging from KMC are assigned to each point in the control space as the targets of the supervised training. This converts the control space phase diagrams into a *speckle plot* of 0's and 1's in the hysteretic region. This binary map that is then fed into the supervised classifiers, which convert it into a smooth map of locking probabilities. The input vector space of the training consists of the values of the control parameters $(\tilde{\psi}_w, \Omega_0)$ at each grid point in the control space. Thus, we have 200×200 samples with two features $(\tilde{\psi}_w, \Omega_0)$ each, and 40000 binary target values that enter the cost function of the classifiers.

The three algorithms chosen to estimate $p(L|\tilde{\psi}_w, \Omega_0)$ are support-vector machines (SVM), logistic regression (LR), and a fully-connected feed-forward neural network (NN). The support-vector machine uses a nonlinear kernel in the form of a radial basis function (RBF) and the logistic regression uses a rational function or *Padé approximants* for the argument of its sigmoid function. These nonlinearities are introduced to deal with difficulties associated with shape of the hysteretic region, especially near the CP. The neural network contains four hidden layers, each consisting of hundreds of nodes.

To assess the accuracy of these estimates of $p(L|\tilde{\psi}_w, \Omega_0)$, a 'ground truth' (GT) locking probability is also computed by a direct, but quite expensive, Monte Carlo method, where the ODE's at each point in control space are integrated for $N_i = 10000$ different random initials conditions. The locking probability at each point is then given by the ratio of the tally of the locked points to N_i . Note that more realistic systems will not grant us the luxury of knowing the GT. The reader should also bear in mind the cost of the GT calculation for the ODE system, which takes approximately a week on several dozen processors, in contrast to any of the three MLC's that take a minute each on a single processor. The orders-of-magnitude gain in the computation time makes it possible for one or several of these MLC's to be employed in real-time operations.

Of the three MLC's that have been used, the NN produces the most accurate probabilities. Trials for optimizing the NN suggest that the most accurate results are obtained for an architecture with four hidden layers, each containing 50–200 nodes. The other two methods, RBF SVM and rational LR do not perform as well in capturing the finer features near the CP and along the boundary segments of the hysteretic region.

This manuscript is organized as follows: the analysis and solutions of the ODE model are presented in Section II. The results from the MLC's for the biased case is presented in Section III, with the unsupervised classification results appearing in Section III B. The locking probabilities calculated by the supervised classifiers, as well as the GT probability, appear in Section III C. Section IV mirrors the structure of Section III for the neutral case. A summary and discussion of the results as well as future directions are provided in Section V. Appendix A presents an analogy between the equations for the locking-unlocking bifurcation diagram and those related to the equation of state for a VDW gas. Appendix B describes the unsupervised and supervised MLC's

that are used in this work.

II. THE MODEL: COUPLED ODE SYSTEM OF LOCKING/UNLOCKING BIFURCATIONS

The coupled ODE system derived here represents the interaction of the magnetic perturbation with the error field in the presence of plasma rotation. This system is sufficient to describe the well-known characteristic bifurcations involving the locking and unlocking of a rotating plasma. The advantage the simple model provides is that it can produce tens of thousands of solutions in a few minutes that can then be used to train the ML algorithms. The lessons learned from such a process might be helpful in designing a ML approach for getting the best use out of a simulation or experimental campaign, where the data will necessarily be more sparse.

We begin with a linear response model for the complex-valued reconnected flux $\tilde{\psi}_c$, as described in Refs. 14,15. For simplicity, we assume that the tearing layer is in the constant- ψ VR regime, without the complications of pressure gradient and curvature in the tearing layer¹⁶. The time-dependent linear homogeneous ODE for $\tilde{\psi}_c$ takes on the following form:

$$\left(\frac{d}{dt} + i\Omega_t - \Delta_1 \right) \tilde{\psi}_c = l_{21} \tilde{\psi}_w, \quad (1)$$

where Ω_t (real-valued) is the instantaneous plasma rotation at the rational surface, l_{21} is an inductance-like factor related to the geometry and the current density profile, and Δ_1 is the intrinsic stability parameter.

The angular momentum of the plasma in the tearing layer at the rational surface is subject to a Maxwell torque as well as a restoring force due to the momentum source Ω_0 ^{1,13}:

$$I \frac{d}{dt} \Omega_t = -\text{Im}(\tilde{\psi}_c^* [\tilde{\psi}'_c]) + \mu(\Omega_0 - \Omega_t) \quad (2)$$

where I is the moment of inertia of the tearing layer near the rational surface and the first term on the right represents the Maxwell torque, with the jump in the derivative of the reconnected flux at the rational surface (proportional to the current density there) given by $[\tilde{\psi}'_c] = \Delta_1 \tilde{\psi}_c + l_{21} \tilde{\psi}_w$. The parameter μ represents a physical drag term, related to viscosity of the plasma¹. We write $\tilde{\psi}_c = \tilde{\psi}_t e^{i\theta_t}$, i.e. $\tilde{\psi}_t$ is the amplitude $|\tilde{\psi}_c|$ and θ_t is its phase. This leads from Eq. (2) to

$$I \frac{d}{dt} \Omega_t = l_{21} \tilde{\psi}_w \tilde{\psi}_t \sin \theta_t + \mu(\Omega_0 - \Omega_t). \quad (3)$$

Splitting Eq. (1) (multiplied by $e^{-i\theta_t}$) into its real and imaginary parts yields the coupled ODE system of Eq. (3) with

$$\frac{d}{dt} \tilde{\psi}_t = \Delta_1 \tilde{\psi}_t + \tilde{\psi}_w \cos \theta_t, \quad (4)$$

$$\frac{d}{dt} \theta_t = -\Omega_t - \frac{\tilde{\psi}_w \sin \theta_t}{\tilde{\psi}_t}, \quad (5)$$

where we choose $I = 1$ and $\tilde{\psi}_w \rightarrow l_{21} \tilde{\psi}_w$ for convenience¹⁷. We normalize the time to a nominal tearing time, and fix

$\Delta_1 = -0.1$ and $\mu = 0.01$. In these units $1/\mu$ is the time scale for momentum transport without electromagnetic torques. The choice of Δ_1 indicates a weakly stable spontaneous tearing mode. The value of μ is chosen such that the growth of the mode (in response to $\tilde{\psi}_w$) with this value of Δ_1 proceeds on a faster time scale than $1/\mu$; the ordering $\mu \ll |\Delta_1|$ is consistent with observed modes in an experiment^{18,19}.

The above ODE system requires three initial conditions for $\tilde{\psi}_t$, θ_t and Ω_t . The solutions depend on two parameters we vary, the error field $\tilde{\psi}_w$ and Ω_0 (keeping the remaining parameters fixed). We refer to $(\tilde{\psi}_w, \Omega_0)$ as the *control parameters*, and we assume that experimentalists may have some degree of control over these two parameters. We refer to the trio $(\tilde{\psi}_t, \theta_t, \Omega_t)$ in the time-asymptotic state as the *order parameters*, characterizing the state of the system.

The system of equations in Eqs. (3)-(5) always lead to a steady time asymptotic state. Note that when $\tilde{\psi}_w$ is zero, the system converges to $\tilde{\psi}_t = 0$ and $\Omega_t = \Omega_0$ for $t \rightarrow \infty$; for large Ω_0 and moderate $\tilde{\psi}_w$, the system converges to a state with $\Omega_t \lesssim \Omega_0$ and $\tilde{\psi}_t$ small (unlocked), while for more moderate Ω_0 and large $\tilde{\psi}_w$ the system may converge to a state with low $\Omega_t \ll \Omega_0$ with $\tilde{\psi}_t$ large (locked).

The data required for training the MLC's in Sections III and IV are obtained by integrating Eqs. (3)-(5). However, some insight can be gleaned by analytically examining the time-asymptotic solutions of the system, which are steady state and found by setting the left side of Eqs. (3)-(5) to zero, which yields:

$$\tilde{\psi}_t = \frac{\tilde{\psi}_w}{\sqrt{\Delta_1^2 + \Omega_t^2}}, \quad (6)$$

$$\tan \theta_t = \frac{\Omega_t}{\Delta_1}, \quad (7)$$

$$\mu(\Omega_t - \Omega_0) = -\Omega_t \frac{\tilde{\psi}_w^2}{\Delta_1^2 + \Omega_t^2}. \quad (8)$$

Equation (8) shows the possibility of either one (real) root for Ω_t or three, and Eq. (7) shows $-\pi/2 < \theta_t < 0$ and $0 < \Omega_t < \Omega_0$. For small Ω_t , specifically $\Omega_t \ll |\Delta_1|$, we have $\tilde{\psi}_t \rightarrow \tilde{\psi}_w/|\Delta_1|$ and $\theta_t \rightarrow 0$, i.e., the maximum response to the error field. For large Ω_t , specifically $|\Delta_1| \ll \Omega_t < \Omega_0$, we have $\tilde{\psi}_t \rightarrow \tilde{\psi}_w/\Omega_t$, $\Omega_t \lesssim \Omega_0$, and $\tan \theta_t \approx -\Omega_0/|\Delta_1| \ll -1$, leading to $\theta_t \rightarrow -\pi/2$.

Equation (8), representing the well-known torque balance in steady-state^{1,2}, can be recast as a cubic equation for Ω_t :

$$\Omega_t^3 - \Omega_0 \Omega_t^2 + \left(\Delta_1^2 + \frac{\tilde{\psi}_w^2}{\mu} \right) \Omega_t - \Omega_0 \Delta_1^2 = 0. \quad (9)$$

The locked/unlocked phases of a rotating plasma interacting with an error field have traditionally been described in terms of the solutions of this equation, where, loosely, $\Omega_t/\Omega_0 \ll 1$ means locked and $\Omega_t/\Omega_0 \approx 1$ means unlocked. This cubic equation has real coefficients, and the conditions to have three real roots or one real root are well known. (These equations can be alternately expressed in terms of a cubic equation for $\tilde{\psi}_t^2$, with the same condition for real roots, and a third equation for θ_t can also be solved.)

As we shall discuss further, in the case of a single real root, the distinction between locked and unlocked phases is subject to interpretation. In the case with three real roots for Ω_t , the largest root corresponds to the unlocked state and the smallest root the locked state. These two roots are the attractors while the middle root is not, and is thus physically unobservable, as the initial conditions near the middle root converge to one of the two other roots. This region in control parameter space $(\tilde{\psi}_w, \Omega_0)$ where there are two stable equilibrium states is the region of *hysteresis*. At the boundary of this hysteretic region an abrupt jump (a bifurcation) can occur between locked and unlocked phases. This is analogous to the jump that an order parameter undergoes during a first order phase transition. In fact, the system described by Eq. (9) is analogous to the Van der Waals equation of state for a non-ideal gas. This equation is also cubic in the number density of the gas n , which is the order parameter of the system. We elaborate on this analogy in Appendix A.

We integrate Eqs. (3)-(5) over $t = [0, 10^4]$ for a uniform 200×200 grid of the control parameters $(\tilde{\psi}_w, \Omega_0)$, spanning $\tilde{\psi}_w = [0, 0.2]$ and $\Omega_0 = [0, 5]$. To deal with the issue of the probability of locking, especially in the hysteretic regime, a *single* initial condition $(\psi_t(0), \theta_t(0), \Omega_t(0))$ is chosen randomly for each pair of control parameters $(\tilde{\psi}_w, \Omega_0)$. The size of the space of initial conditions affects the system's tendency to lock, with some initial conditions biased toward locking and others less biased, as explained in the following Sections.

III. THE LOCKING-BIASED CASE

A. The solutions of the ODE's

To bias the solution inside the hysteretic regime toward locking the range for $\tilde{\psi}_t(0)$ is augmented to $[0, 20]$, while that for $\Omega_t(0)$ is reduced to $[0, 1]$. The initial condition for the phase $\theta_t(0)$ is always sampled over $[0, 2\pi]$, regardless of the locking bias.

The ODE solutions solved on the 200×200 grid of control parameters are shown in Fig. 1a, where two of the three order parameters $(\tilde{\psi}_t, \Omega_t)$, are plotted as a scatter plot, while θ_t is ignored for convenience at this stage. These two order parameters form an 'L'-shaped pattern in the solution space, where the vertical 'arm' corresponds roughly to unlocked solutions and the horizontal arm to locked solutions, but the density of points is not noticeable lower near the origin.

As argued in the last section, the locked states have $\tilde{\psi}_t \lesssim \tilde{\psi}_w/|\Delta_1|$, $\Omega_t \gtrsim 0$, and $\theta_t \approx 0$, whereas the unlocked states have $\tilde{\psi}_t \gtrsim 0$, $\Omega_t \lesssim \Omega_0$, and $\theta_t \approx -\pi/2$. Motivated by these observations, we define the *normalized order parameters* $\tilde{\psi}_n \equiv |\Delta_1| \tilde{\psi}_t / \tilde{\psi}_w$ and $\Omega_n \equiv \Omega_t / \Omega_0$, and discuss the advantages of using them rather than the raw order parameters. Because we have $-\pi/2 < \theta_t < 0$, it is not necessary to scale θ_t further. In terms of these normalized parameters, locked states have $\tilde{\psi}_n \sim 1$, $\Omega_n \sim 0$ and $\theta_t \sim 0$; whereas unlocked states have $\tilde{\psi}_n \sim 0$, $\Omega_n \sim 1$ and $\theta_t \sim -\pi/2$. From Eqs. (6) and (7), we have $\tilde{\psi}_n = \cos \theta_t$, a direct relationship inde-

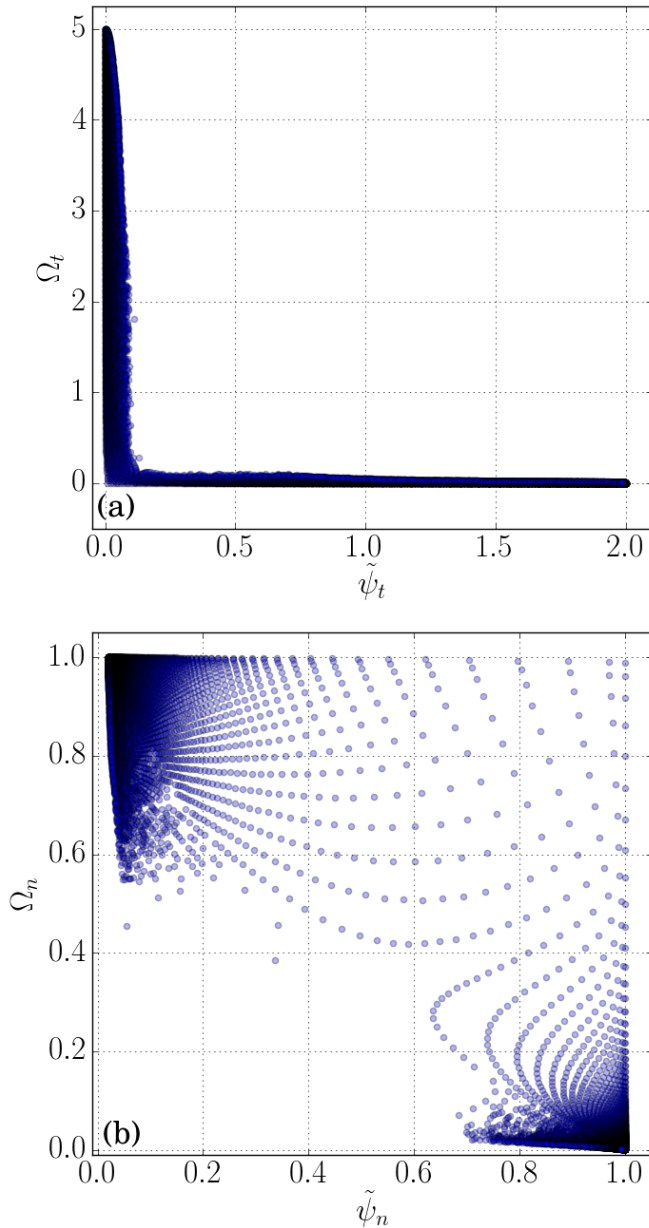


FIG. 1. Scatter plots of the time-asymptotic solutions of the ODE's: (a) the *raw* (unnormalized) order parameters $\tilde{\psi}_t$ and Ω_t , and (b) their *normalized* counterparts $\tilde{\psi}_n$ vs Ω_n . In (b) the solutions appear to cluster about the points $(\tilde{\psi}_n, \Omega_n) = (1, 0)$ and $(0, 1)$, corresponding to locked and unlocked time-asymptotic states, respectively.

pendent of the control parameters $(\tilde{\psi}_w, \Omega_0)$. For this reason, when normalized order parameters are used, the variable θ_t becomes redundant. In contrast, the relation between $\tilde{\psi}_n$ and Ω_n depends on the control parameter Ω_0 . It is important to note that the redundancy of θ_t also reduces the dimensions of the order parameter space from 3D to 2D. In terms of the remaining normalized order parameters it becomes far less ambiguous what constitutes a locked or an unlocked state of the plasma. On the unit square in $(\tilde{\psi}_n, \Omega_n)$ shown in Fig. 1b, the locked states are clustered near the NW corner, and the

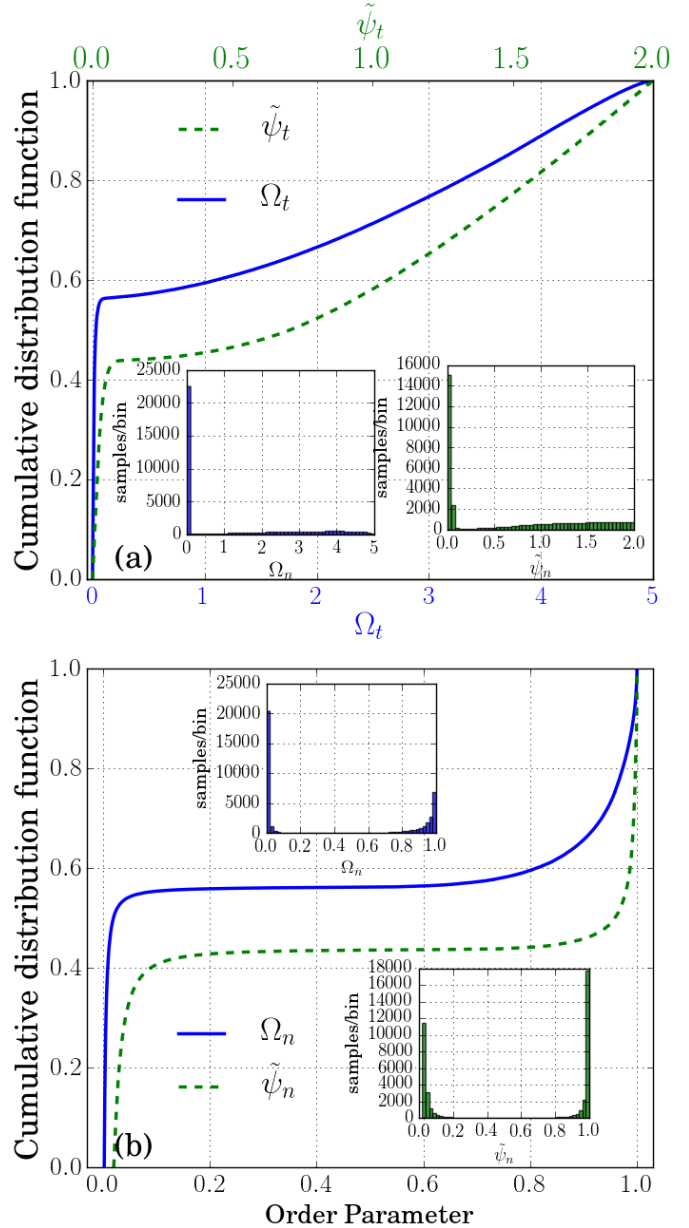


FIG. 2. The cumulative distribution functions (CDF's) of the (a) *raw* order parameters $\tilde{\psi}_t$ (dashed green) and Ω_t (solid blue); and (b) *normalized* order parameters $\tilde{\psi}_n$ (dashed green) and Ω_n (solid blue). The histograms (i.e., estimates of the marginal probability density distribution) in the insets show the number of samples per bin with respect to each of the two order parameters.

unlocked states near the SE corner; more than 99% of the solutions fall into these two clusters.

There is a small population of non-clustering points that appears in the center of Fig. 1b. However, the concentration of clustering points near the two corners is several orders of magnitude greater than that of these non-clustering points, as evidenced by the darkening of the color (indicating the density of points) in Fig. 1b. As we will see, this clustering due to the choice of normalization aids in classification greatly. Also, with this normalization choice, with param-

ters on the unit square, it is natural to use a Euclidean metric tensor for distances, used in the unsupervised classification of Section III B²⁰.

The crucial role played by normalization of the order parameters is further illustrated in Figs. 2a and b. Figure 2a shows the 1D cumulative distribution functions (CDF) of the time-asymptotic raw order parameters $\tilde{\psi}_t$ (dashed green) and Ω_t (solid blue), while Fig. 2b shows the 1D CDF of the time-asymptotic normalized order parameters $\tilde{\psi}_n$ (dashed green) and Ω_n (solid blue). The figures also include the probability density function (derivative of the CDF) shown as histograms in the insets, with the blue bars labeling Ω_t and Ω_n , and the green bars labeling $\tilde{\psi}_t$ and $\tilde{\psi}_n$. The CDF's of the raw order parameters in Fig. 2a show a tendency toward clustering around zero for both Ω_t and $\tilde{\psi}_t$, but vary noticeably for intermediate values of these parameters. The jumps in Ω_t and $\tilde{\psi}_t$ correspond roughly to the horizontal and vertical wings of the scattered solutions shown in Fig. 1a, respectively.

The cumulative distribution functions are re-plotted as a function $\tilde{\psi}_n$ and Ω_n in Fig. 2b to demonstrate the dramatic effect of our choice of normalization of the solutions. The clustering behavior, described above for Fig. 1b, re-emerges here as well. The CDF traces now evince two areas of fast increase: one around $\Omega_n \lesssim 0.05$ ($\tilde{\psi}_n \gtrsim 0.9$) and another around $\Omega_n \gtrsim 0.8$ ($\tilde{\psi}_n \lesssim 0.1$). There is very little increase in the CDF for intermediate values, showing that the number of intermediate solutions between the locked and unlocked states are very few. Similarly, the histograms shown in the upper inset for Ω_n and lower inset for $\tilde{\psi}_n$ feature two spikes located at $\Omega_n \approx 0$ ($\tilde{\psi}_n \approx 1$) and $\Omega_n \approx 1$ ($\tilde{\psi}_n \approx 0$), respectively. Thus, the normalization makes it clear that the majority of the population clusters into two groups: one rotating very slowly with the magnetic perturbation $\psi_t \simeq \psi_w/|\Delta_1|$ (locked) and the other rotating fast at nearly Ω_0 with $\tilde{\psi}_t \approx 0$ (unlocked). The minimum of the histograms, corresponding to the flattest part of the CDF, suggest the possibility of a threshold for locking in terms of a single order parameter. These minima indicate locking for $\Omega_n \sim 0.5$ or $\tilde{\psi}_n \sim 0.5$. Both criteria result in approximately same rate of locking: 56% of the total population is locked for this particular range of control parameters. We will compare the locking criteria based on the CDF's to those from the classification results in Sec. III B.

Phase diagrams, i.e. the time-asymptotic solutions as a function of the control parameters $(\tilde{\psi}_w, \Omega_0)$ on a 200×200 uniform grid are shown in Fig. 3. Here, (a) and (b) show color plots of the time asymptotic *normalized* rotation frequency Ω_n and mode amplitude $\tilde{\psi}_n$, respectively. As discussed, θ_t is redundant and therefore omitted. The phase diagrams show three distinct regions: two that are solely occupied with solid red or blue pixels and a third region, the hysteretic region, with an intermixing of the blue and red pixels. The regions of solid color correspond to the cubic equation having one real root. The solid blue represents cases with $\Omega_n \ll 1$, $\tilde{\psi}_t/\tilde{\psi}_w \approx 1$ in Figs. 3a/3b, i.e. unlocked solutions. These are the solutions that are hardly influenced by the error field. The solid red represents cases with $\Omega_n \approx 1$, $\tilde{\psi}_t/\tilde{\psi}_w \ll 1$ in Figs. 3a/3b, i.e. the locked so-

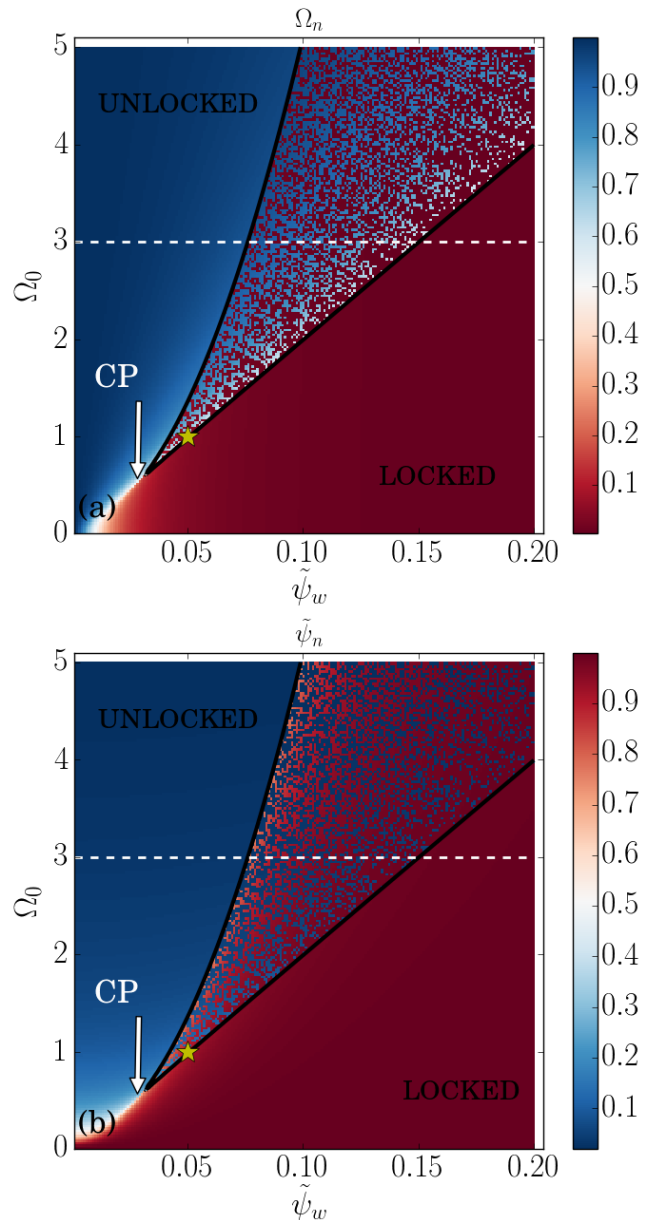


FIG. 3. The phase diagrams: Time-asymptotic solutions of the ODE's in the control space $(\tilde{\psi}_w, \Omega_0)$, for the case that is biased toward locking. The color in (a) represents the normalized rotation frequency $\Omega_n \equiv \Omega_t/\Omega_0$, and in (b) the normalized mode amplitude $\tilde{\psi}_n \equiv \tilde{\psi}_t|\Delta_1|/\tilde{\psi}_w$. Note the color bar is flipped in (a) and (b) to have a similar coloring scheme for both panels. The two black curves bound the hysteresis region that merge at the critical point (CP) of the system. The white dashed horizontal line is a cut along which we illustrate the hysteretic behavior in 1D in Fig. 5. The significance of the gold star is explained further below.

lutions. These are the solutions that are strongly influenced by the error field. The third region, squeezed between the solid blue and red region, where the pixels with a spectrum of colors ranging from red to blue are intermixed, corresponds to the hysteretic region. This is where the analytic solution of the torque balance, Eq. (9), yields three real roots. In this

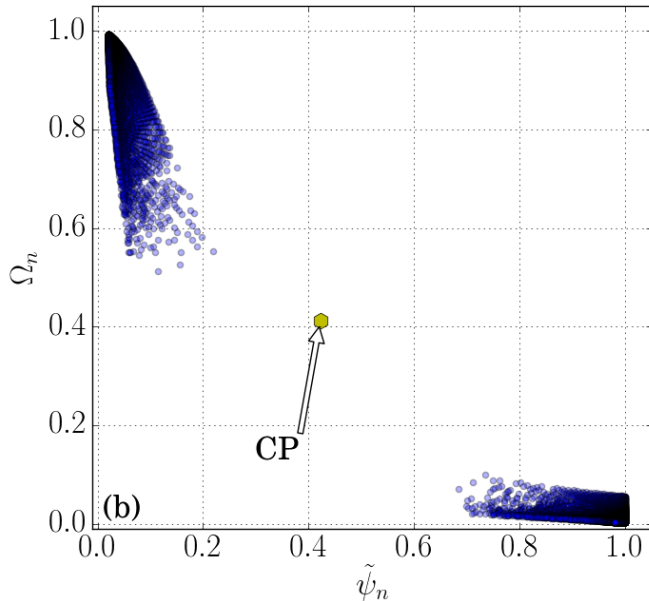


FIG. 4. The scatter plot of Fig. 1b where most of the smoothly-transitioning points below the CP have now been eliminated by keeping only the solutions that are generated with $(\tilde{\psi}_w, \Omega_0) \geq (0.05, 1.0)$. This point is marked by the gold stars of Figs. 3a and b just inside the hysteretic region. The critical point (CP) order parameters of the system are also shown.

region, there can be two nearby points in the control space, randomly initialized with two different values of the order parameters, with one going to a locked state and the other going to an unlocked state, and hence the ‘speckling’ pattern.

The speckles apparent to the eye in Fig. 3 depend on the particular value of $(\tilde{\psi}_w, \Omega_0)$ as well as on the range of the initial conditions of the randomly chosen order parameters and the actual realization of these random variables. On the other hand, the density of speckles is insensitive to the different realization of the random initial conditions so long as the range of the initial conditions is kept the same from one execution of the ODE system to the other. Thus, the location of the individual speckles inside the hysteretic region in the control space corresponds to a microscopic quantity, sensitive to fluctuations; while the density of the speckles therein corresponds to a more robust macroscopic quantity. It is this density that the MLC’s used in this work estimate in Sections III C and IV B.

Above the upper black curves appearing in Fig 3 the cubic has only unlocked states; below the lower black curve the cubic has only unlocked states. These curves bound the hysteretic region. These two bifurcation boundary segments merge at the *critical point* (CP) of the system, situated at $(\tilde{\psi}_w, \Omega_0) = (0.0283, 0.520)$. Near the CP, the two curves meet at a tangency, $\delta\Omega_0 \propto (\delta\tilde{\psi}_w)^{3/2}$. This point will be of importance when we deal with supervised classification to calculate the probability of locking in this region. Below and to the left of the CP, one state of the plasma can smoothly transition into another without undergoing bifur-

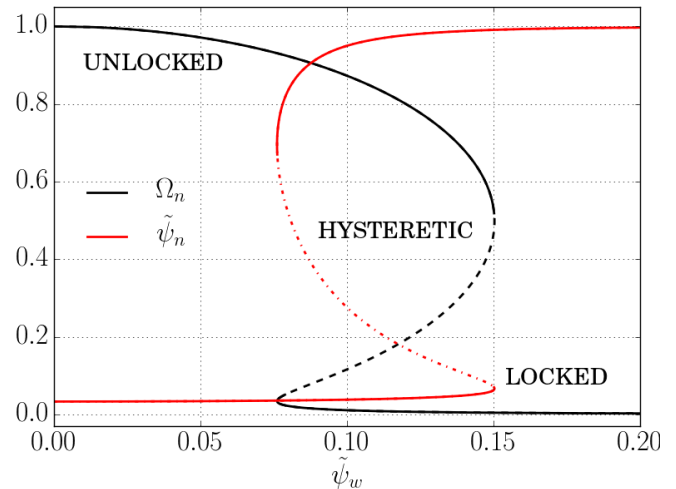


FIG. 5. Bifurcation in the ODE solution represented separately by the normalized order parameters Ω_n and ψ_n . These are shown as functions of $\tilde{\psi}_w$ at $\Omega_0 = 3.0$ (dashed horizontal white lines in Figs. 3a and b.) The upper/lower part of the $\Omega_n/\tilde{\psi}_n$ curve corresponds to unlocked states; the lower/upper part of the $\Omega_n/\tilde{\psi}_n$ corresponds to locked states. In the hysteretic region for $0.08 < \tilde{\psi}_w < 0.15$, both locked and unlocked states exist. The dashed parts of the traces represent the ‘forbidden’ intermediate state, which is not observed.

cations, much like continuously transitioning between the gaseous and liquid phases of a VDW gas (Appendix A). Note the difference in the color pattern below the CP for the two plots. The widely scattered points between the two clusters in Fig. 1b correspond exactly to these smoothly-transitioning points. These smoothly-transitioning points can be eliminated by re-plotting only the solutions that are generated from the control parameters that lie above the CP, in this case specifically for $(\tilde{\psi}_w, \Omega_0) \geq (0.05, 1.0)$, which is marked by the gold star in Fig. 3. This filter produces the two well-separated clusters shown in Fig. 4. Another related point is that the states in the vicinity of the CP have small values of Ω_n and especially $\tilde{\psi}_n$ and therefore the occurrence of disruptions due to locking is not such a serious concern.

The dashed horizontal white lines in Figs. 3a and b represent a 1D cut at $\Omega_0 = 3.0$ along which we further illustrate the hysteresis inherent in our ODE system. See Fig. 5, which shows the bifurcation in the solutions Ω_n and $\tilde{\psi}_n$ as a function of $\tilde{\psi}_w$ for $\Omega_0 = 3.0$. The solid black (Ω_n) and red ($\tilde{\psi}_n$) traces are the analytic curves that are extracted by solving the time-asymptotic solutions given in Eqs.(6)–(8). The upper branch of the red curve and the lower branch of the black curve correspond, respectively, to locked states; the lower branch of the red curve and the upper branch of the black curve correspond to the unlocked states. The ‘forbidden’ solutions in the hysteretic region are represented by the dashed portions of the traces.

In the following sections, we employ a two-step strategy, with each stage leveraging machine-learning classifiers (MLC’s) in a particular way. Stage 1 entails an unsupervised classification in terms of the normalized order parameters of the ODE as locked or unlocked. This step uses *K-means clus-*

tering (KMC), described in Appendix B. This gives us an unambiguous locking criterion in the normalized order parameter space, $(\tilde{\psi}_n, \Omega_n)$. As discussed above, the normalization of the solution space onto a unit square facilitates the task of classification. In more complex systems, e.g. in numerical simulations, it may not be possible to do such a scaling to aid the classification process; in such cases it will be important to use unsupervised classifiers to determine the class of each solution. In stage 2, we use the supervised MLC's to calculate the conditional probability of locking within the hysteretic region in the control parameter space. For this process we use the control parameters $(\tilde{\psi}_w, \Omega_0)$ as the input vectors instead of the normalized order parameters, and the sample classes labeled according to KMC as the target values, to train the MLC's. This ML-based probability is then compared to a Monte Carlo-type calculation of probability, described in Section III C, which we call the *ground truth probability*.

B. Classifying the ODE solutions as locked or unlocked

The main classification result of this section is based on using the normalized order parameters, $\tilde{\psi}_n$ and Ω_n as the input for the unsupervised classification. We refer to this procedure as a 2D classification. The classification is also repeated in 1D, using either Ω_n or $\tilde{\psi}_n$ as the single input. The results show the 2D classification to be only marginally better than the 1D classification based on either Ω_n or $\tilde{\psi}_n$ alone. This is another indication of the fact that the normalizations (compare Figs. 1a and b) facilitate this task of unsupervised learning greatly, by redistributing most of the samples into the two aforementioned clusters.

We use *K-means clustering*^{21,22} (KMC) to classify each solution of the ODE's as locked or unlocked in an unsupervised fashion, i.e. without human input. The details of this algorithm are described in Appendix B 1. Briefly, KMC geometrically partitions N sample points into K clusters by computing the centroid or cluster center of mass, and then assigning each sample to the cluster with the nearest centroid. (This classification in terms of distance shows the importance of picking an appropriate metric in the order parameter space.) KMC is a technique that is used in image segmentation and compression.

The results of the classification by K-means clustering trained on the values of $(\tilde{\psi}_n, \Omega_n)$ are displayed in Fig. 6. The blue dots represent unlocked cases and the red crosses their locked counterparts. The centroids, marked by two large yellow diamonds, are located at $(\tilde{\psi}_n, \Omega_n) = (0.98, 0.01)$ and $(0.05, 0.94)$ for the locked and unlocked states, respectively. Since the two cluster centers are located in the upper left and lower right corners, KMC classification in this case produces almost a perfectly diagonal decision boundary (dashed line), separating the locked and unlocked populations. This means any ODE solution with $\tilde{\psi}_n \geq \Omega_n$ corresponds to a locked phase of the plasma. This criterion indicates 55% of the population to be locked, in good agreement with the results based on the minimum derivative of the cumulative

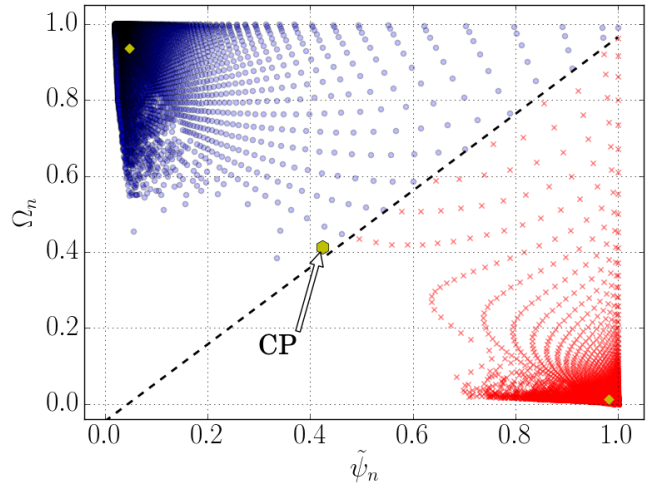


FIG. 6. The unsupervised classification of the results of Fig. 1 into locked (red crosses) and unlocked (blue dots) classes via K-means clustering performed on the pair of normalized order parameters $(\tilde{\psi}_n, \Omega_n)$. The yellow markers near the corners indicate the centroids of the clusters, and the critical point (CP) is labeled.

distribution functions shown in Fig. 2b.

For the classifications based on a single input consisting of the values of $\tilde{\psi}_n$ (or Ω_n) as the training input, KMC splits the population into two classes by a vertical (respectively, horizontal) line near the middle of the range for each order parameter: The classification based on $\tilde{\psi}_n$ indicates locking for $\tilde{\psi}_n \gtrsim 0.51$ and classification based on Ω_n indicates locking for $\Omega_n \lesssim 0.48$. These locking thresholds are in good agreement with the results based on the minimum derivative of the cumulative distribution functions shown in Fig. 2b. The two individual criteria in this case lead nearly to the same number of locked states, 56%, as the decision boundary based on both $(\tilde{\psi}_n, \Omega_n)$. The reason for such a close agreement between the locking tallies in spite of three seemingly different criteria for locking, is the fact that the widely scattered data in Fig. 6 are very sparse.

The KMC results discussed here have been found to agree well with another unsupervised classifier called Gaussian-mixture models (GMM), which fit K -many anisotropic Gaussians to a data set, where each Gaussian represents one cluster. Unlike KMC, which is a geometric method that makes *hard* assignments, GMM are a probabilistic method that makes *soft* assignments to the data points, and can account for the cluster density. In fact, the cluster density is one of the parameters that the GMM algorithm optimizes. However, since the data in use here have nearly the same density for both clusters, the classification results with GMM are very similar to those with KMC. Thus, we proceed with the KMC classification results for the remainder of the manuscript.

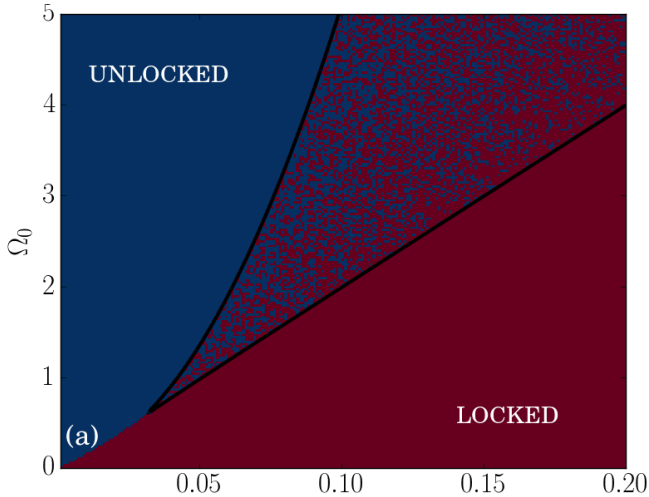


FIG. 7. The intermediate processing step between unsupervised and supervised classifications: the KMC classification results are used to convert the phase diagrams of Fig. 3 to a binary map of 0's and 1's by marking any order parameter with $\tilde{\psi}_n > \Omega_n$ locked (red crosses) and others with $\tilde{\psi}_n \leq \Omega_n$ as unlocked (blue dots). We call this the 'speckling diagram'.

C. Calculating Locking Probabilities via Supervised Classification

In stage 2, we use the supervised classifiers described in Appendix B 2 to calculate the probability of locking $p_L \equiv p_L(L|\tilde{\psi}_w, \Omega_0)$, conditional on the control parameters of our model in the hysteretic region of the control space. This involves switching from the 2D space of normalized order parameters $(\tilde{\psi}_n, \Omega_n)$ of the previous section to the 2D space of the control parameters $(\tilde{\psi}_w, \Omega_0)$, appearing as the 200×200 uniform grid in Figs. 3a and b. We then assign values from $t \in \{0, 1\}$ (unlocked, locked, respectively), to each control space point $(\tilde{\psi}_w, \Omega_0)$ according to the classification of the previous section. This conversion is done by imposing the KMC classification all of the 200×200 points on this grid whereby each point with $\tilde{\psi}_n > \Omega_n$ is classified as locked (red) and each point with $\tilde{\psi}_n \leq \Omega_n$ is classified unlocked (blue). Figure 7 shows the result of this procedure: a speckling pattern within the hysteretic region emerges. It is this binary map that is fed into the supervised classifiers in this section, which convert it into a smooth map of locking probabilities $p_L = p(L|\tilde{\psi}_w, \Omega_0)$.

The justification for using a single speckling diagram is related to the aforementioned insensitivity of the macroscopic quantities such as the density and number of locked states within the hysteretic region to the different realizations of the randomly chosen initial conditions. This insensitivity has been confirmed by solving the ODE's on the same control grid five times with five different realizations of the initial conditions and observing the variance in the number of locked states within the hysteretic region to be $\sim 1\%$. Thus, it suffices to generate only a single speckling map, as shown in Fig. 7.

To gauge the accuracy the conditional probabilities calcu-

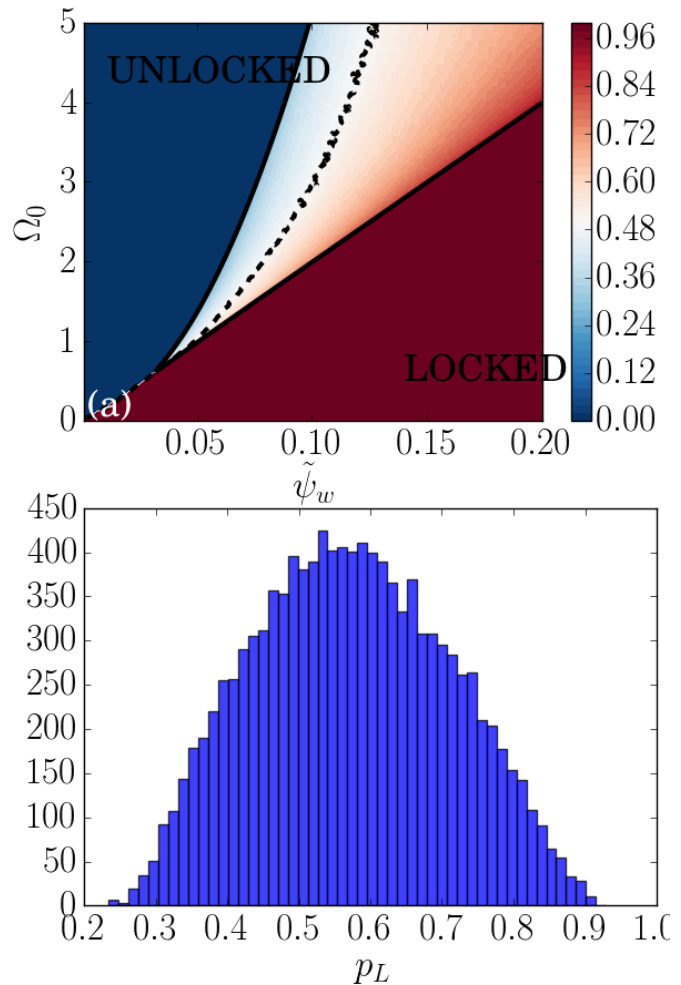


FIG. 8. (a) Ground truth conditional probability of locking $p_{GT}(L|\tilde{\psi}_w, \Omega_0)$ ($p_L^{(GT)}$ for short) calculated via a basic Monte Carlo approach by solving the ODE's at each point in the control space for $N_i = 10^4$ randomly selected initial conditions, and using the diagonal decision boundary of Fig. 6 as the criterion for determining which of the 10^4 points are locked. (b) A histogram of the $p_L^{(GT)}$ values in the hysteretic region. Note the lack of very low and very high values of $p_L^{(GT)}$ in this distribution.

lated by the MLC's we first calculate the *ground truth* (GT) conditional probability $p_{GT}(L|\tilde{\psi}_w, \Omega_0)$ or $p_L^{(GT)}$ for short, by solving the ODE's repeatedly at each point in the control space for $N_i = 10000$ randomly selected initial conditions. In contrast a single initial condition at each point in the control space is used to generate the ODE solutions presented in Section III A, which make up the data for the MLC's used below. The ground truth probability for locking is calculated by a Monte Carlo-type approach: by accepting each solution satisfying the locking criterion: $\tilde{\psi}_n \geq \Omega_n$ based on the diagonal decision boundary shown in Fig. 6, and rejecting the solutions that fail this criterion. We then take the ratio of the number of accepted (locked) outcomes over the total number of initial conditions used for each of our 200×200 points²³ in the control space to arrive at $p_L^{(GT)}$. The resulting

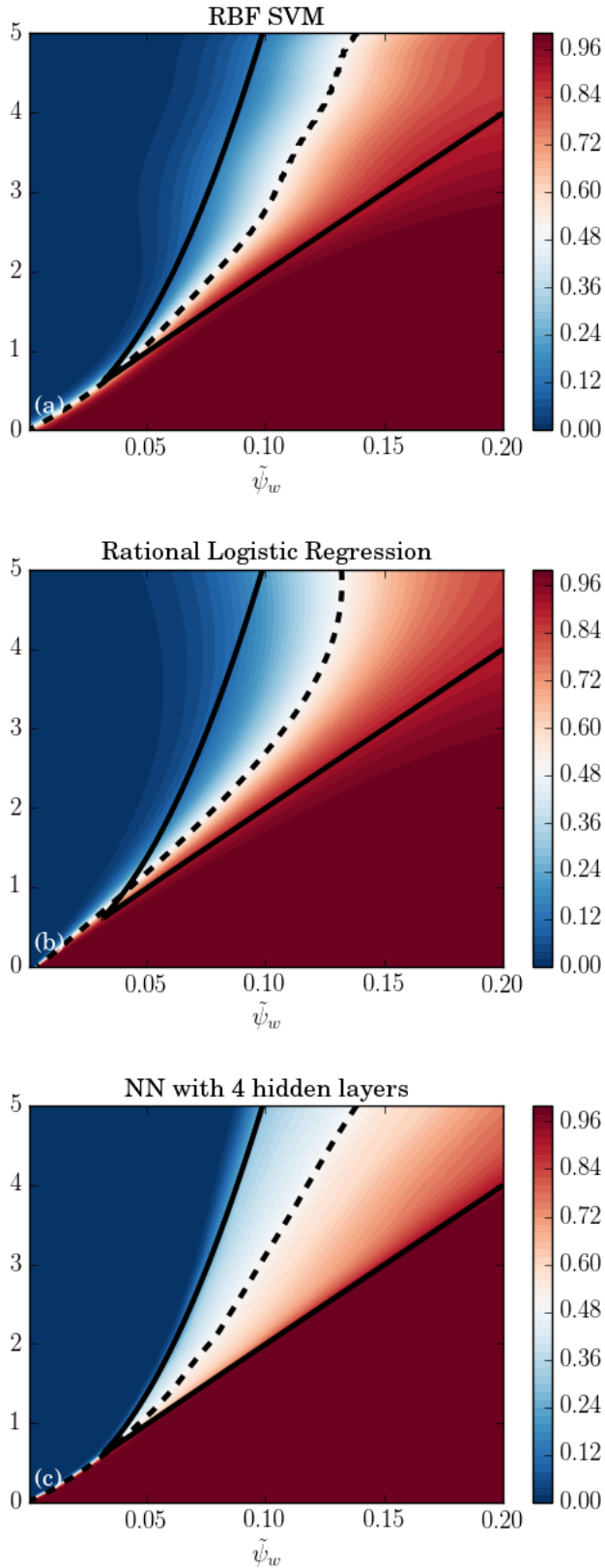


FIG. 9. The contours of the estimated conditional probability of locking as obtained from the MLC’s. Shown are results with (a) a support-vector machine (SVM) with a radial basis function (RBF) kernel, (b) logistic regression (LR) with a rational argument, and (c) a fully-connected feed-forward neural network (NN) with four hidden layers, each consisting of 200, 100, 100, and 200 nodes, respectively. The dashed black curve in all cases marks $p_L = 0.5$.

GT probability is shown in Fig 8a. The color represents the conditional probability p_L , with $p_L = 0$ corresponding to an unlocked state (blue) and $p_L = 1$ corresponding to a locked state (red). The figure also shows the analytic bifurcation boundary segments as thick black lines that demarcate the hysteretic region. As evidenced by the dashed black curve marking $p_L = 0.5$, the contours of low probability are fairly close to the left boundary segment. In other words, the hysteretic region is taken up mostly by a region with a higher probability of locking, consistent with the locking bias introduced into the initial conditions. In a realistic scenario like an experimental or MHD simulation campaign, we will not have the luxury of calculating the ground truth probabilities.

The accumulation of the low probabilities toward the left boundary segment is strongly suggested in Fig. 8b, which shows a histogram of the values of $p_L^{(GT)}$ —a probability density of probabilities—restricted to points in the hysteretic region. This distribution is peaked, with a mean of roughly 0.55 (slightly shifted toward locking) and $0.4 \lesssim p_L^{(GT)} \lesssim 0.7$ in a large area of the hysteretic region, consistent with the large white region in Fig. 8a. The peaked histogram also shows an unexpected and peculiar structure in the form of noticeable gaps in the distribution for $p_L \lesssim 0.25$ and $p_L \gtrsim 0.9$. The distribution of GT probabilities changes significantly for the *neutral* (less biased) case, discussed in Section IV, where a displacement of the large flat region to lower probabilities is observed.

To calculate the ML-based probability, we use the two control parameters $(\tilde{\psi}_w, \Omega_0)$ as the input vector for the training. The target values of the training are the binary classification outcomes shown as the red and blue dots in Fig. 7. Note we do not hold out a fraction of the data for testing as it is commonly done in ML, because we are able to compare with the GT probabilities.

Three supervised classifiers—described in Appendix B 2—are used to estimate p_L and their results are displayed in Fig. 9: panel (a) shows a support vector machine (SVM) with a kernel that uses Gaussian radial basis functions (RBF) panel (b) shows LR with a rational basis function; and panel (c) shows the neural network (NN) with four hidden layers, each consisting of 200, 100, 100, and 200 nodes, respectively. The same color coding as that in Fig. 8a is used, with blue colors indicating low probability of locking and red colors indicating high probability of locking. The dashed black curve marks the $p_L = 0.5$ contour in each panel, and the solid black curves demarcate the hysteresis region. An accurate probability should reproduce $p_L \rightarrow 0$ at the upper (left) boundary segment and $p_L \rightarrow 1$ at the lower (right) boundary segment. The tangency at the critical point $\delta\Omega_0 \propto (\delta\tilde{\psi}_w)^{3/2}$, leading to a thin region over which the probability varies rapidly, presents a serious challenge to the ML algorithms. The error in the ML probabilities is shown in Fig. 10 in terms of a residual, which is the difference between the GT probability and those calculated by each one of the three MLC’s. The residuals are only plotted in regions where $0.01 \leq p_L \leq 0.99$, to focus on the hysteretic region and its immediate surroundings. The root mean square error (RMSE) with respect to the GT probability for the same

region is also reported in each panel of Fig. 10.

Support-vector machine with a radial basis function kernel (RBF SVM) (Fig. 9a) captures the rough structure, including the accumulation of low probability contours toward the left boundary segment and the sharp transition between the locked and unlocked regions below the CP. The latter property is a consequence of imposing a binary classification the smoothly-transitioning solutions. This is also where the largest residuals appear for all three MLC-calculated probabilities (Figs. 10a–c). The general shape of the SVM probability contours do not conform to the shape of the hysteretic region: the $p_L \approx 0$ and $p_L \approx 1$ contours spill grossly outside the bifurcation boundary segments marked by the solid black curves. This is also evident in the residual plot, Fig. 10a, which indicates a RMSE= 0.10 for this classifier. An overfitting²¹ problem in the form of wiggles in the left of the left hysteresis boundary segment and in the upper right corner of Fig. 9a appears as well.

The results from the rational logistic regression (RLR) are shown in Fig. 9b. As described in Appendix B 2, the RLR uses for the argument of the sigmoid a rational function that is a quadratic function of the two inputs (ψ_w, Ω_0) in the numerator and a linear function (ψ_w, Ω_0) in the denominator. Similar to the results of the RBF SVM, the p_L contours as calculated by RLR also spill noticeably outside the hysteretic region in spite of the nonlinear argument of the sigmoid offering greater flexibility for fitting. There is significant error in the vicinity of both bifurcation boundary segments as indicated by the residual plot for RLR shown in Fig. 10b. The root mean square error is 0.11, approximately the same as that reported for RBF SVM, as is the area over which the residual is calculated. The similarity between the RBF SVM and RLR results are likely due to the fact that SVM uses logistic regression to convert the binary classification outcomes into a probability. Both the SVM and the RLR do fairly well in the vicinity of the CP, broadening the probability contours somewhat near the tangency of the two bifurcation curves.

The neural network (NN) shown in Fig. 9c captures the accumulation of the low p_L toward the left boundary segment due to the locking bias, like the previous two classifiers. In terms of other metrics however, the NN clearly outperforms RBF SVM and RLR in accurately computing p_L . The neural network probability p_L contours mostly capture the critical property that $p_L \rightarrow 0/p_L \rightarrow 1$ at upper/lower bifurcation boundary segments; this is shown much better than with the other two classifiers. This is also evident in the residual plot of Fig. 10c that evinces an area that mostly conforms to the shape of the hysteretic region, and conforms well in the vicinity of the CP tangency. The NN results also have significantly smaller residuals than those shown in Figs. 10a–b, producing the smallest RMSE (0.055), as indicated by Fig. 10c. However, what stands out the most about the NN’s performance is the powerful smoothing effect produced by the use of an architecture with several hidden layers, each containing hundreds of nodes. For each pixel, it seems the NN accurately brings in contributions from the neighboring pixels to result in an accurate probability, without doing too much smoothing near the boundary segments and especially

the CP. This smoothing effect breaks down for fewer than approximately fifty nodes in most of the layers. Further details of the NN architecture are discussed in the next paragraph.

The sensitivity of the NN results to the network architecture has been investigated in terms of the number of hidden layers, the number of nodes contained within each hidden layer, and finally the type of nonlinear activation function used for each node. The locking probability exhibits notable sensitivity to these parameters, and the best accuracies are observed for architectures that use at least two hidden layers with at least 50 nodes in each layer. It should be noted that the choice of optimal network architecture has no closed form solution and can only be found with trial-and-error²⁴. The optimization of the NN parameters (weights) is carried out via *backpropagation*, which is discussed in Appendix B 2. Another feature of the NN is the fact that the converged solution does not necessarily correspond to the global minimum, but rather to a local minimum that is reasonably close to the global minimum²⁵.

To illustrate further the differences between the three MLC’s, we take horizontal and vertical slices through the 2D probability plots of Fig. 9 to compare the profiles of the MLC probabilities to those extracted from the GT probability shown in Fig. 8a. The results are shown in Fig. 11: panel (a) shows p_L profiles as a function of ψ_w at $\Omega_0 = 1.5$ (black traces on the left) and 3.8 (red traces on the right); panel (b) shows p_L profiles as a function of Ω_0 at $\psi_w = 0.06$ (black traces on the left) and 0.15 (red traces on the right). One feature that immediately stands out here is the aforementioned gap in the GT probability, which appears here as very sharp increases and even apparent jumps in the solid curves near $p_L = 0$ and 1.0. Neither SVM (dashed-dotted), nor RLR (dotted) is able to capture this peculiar property, while the NN partially reproduces it, albeit a smooth version of it. The neural network also captures the flatness of $p_L^{(GT)}$ in the hysteretic region that is evident in Fig. 9c.

Another important issue addressed here regards the amount of data required for accurate training, as it relates to the sparsity of data in more realistic situations. A high-fidelity simulation campaign with an MHD framework like NIMROD²⁶ may take months, even years to complete. In the case of a large-scale experiment, data can take years to collect^{5,7–9}. Thus, it is imperative to determine the minimum sample size required to train the MLC’s for obtaining accurate predictions. To achieve this, we subject smaller data sets to the same procedure as described above to determine the threshold in sample size at which unacceptable loss of accuracy occurs. We have determined this threshold to be approximately 40×40 samples for the NN. Below this threshold, the RMSE exceeds 0.1 for the NN. The probabilities calculated by SVM and RLR are not as sensitive to sample size as the NN, possibly because of the much smaller number of free parameters (weights) that these algorithms optimize. The requirement of several thousand samples for accuracy is on par with the number of DIII-D shots that went into constructing the disruption avoidance algorithm of Ref. 9.

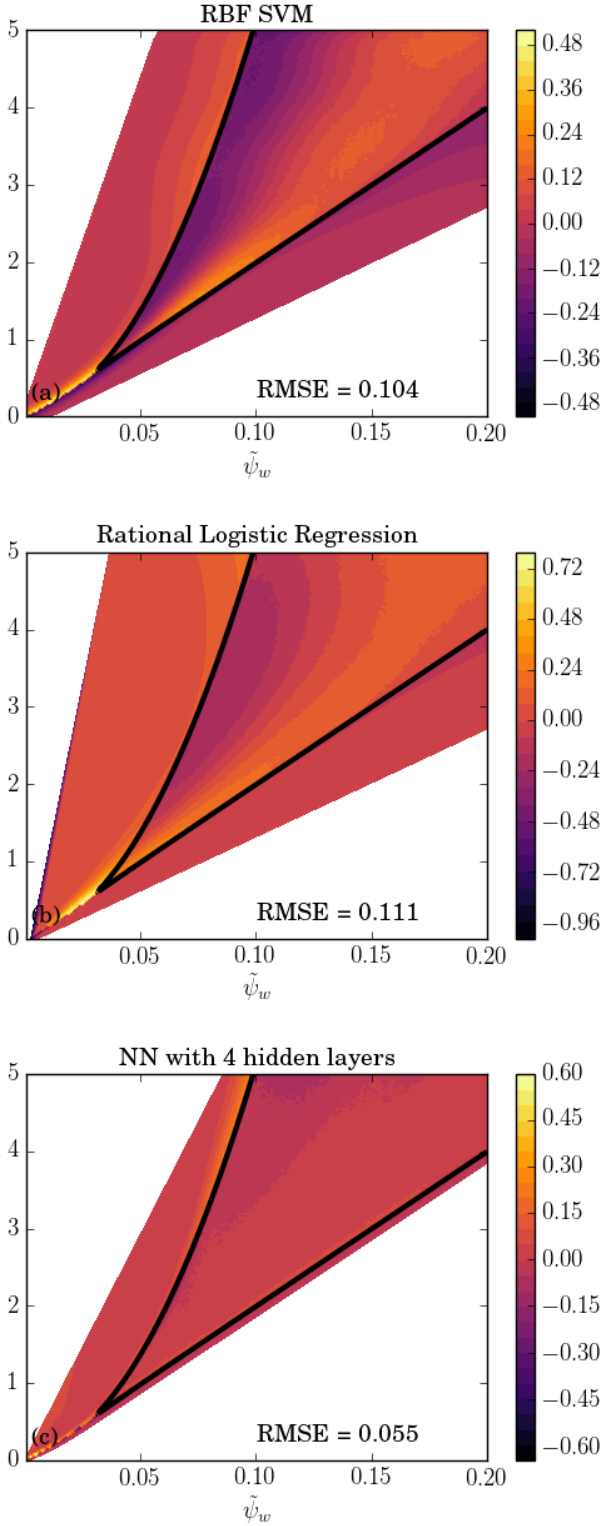


FIG. 10. Residuals between the conditional probability of locking (p_L) obtained from the ML classifiers of Fig. 9 and those obtained from the ground truth Monte Carlo calculation $p_L^{(GT)}$: (a) support-vector machines (SVM) with a radial basis function (RBF) kernel, (b) logistic regression (LR) with a rational argument, and (c) a fully-connected feed-forward neural network (NN) with four hidden layers, each consisting of 200, 100, 100, and 200 nodes, respectively. The residuals are only shown in regions where $0.01 \leq p_L \leq 0.99$, to focus on the hysteretic region and its immediate surroundings. The root mean square error (RMSE) of p_L with respect to $p_L^{(GT)}$ over the same region is also shown.

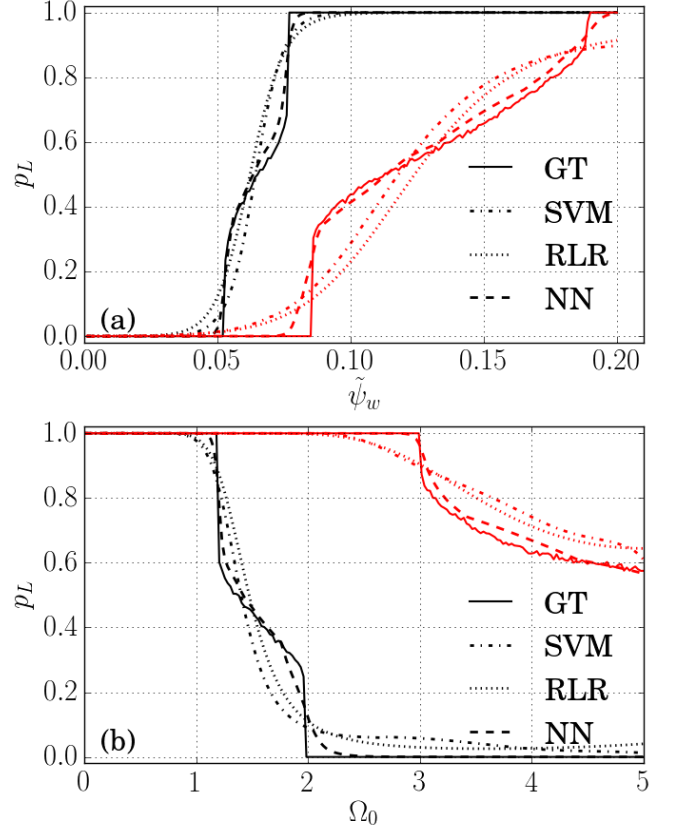


FIG. 11. Profiles of $p_L^{(GT)}$ and p_L obtained by the three MLC's, described in captions of Figs. 9 and 10. Panel (a) shows horizontal slices, i.e. the locking probability p_L as a function of $\tilde{\psi}_w$ at $\Omega_0 = 1.5$ (black traces on the left) and 3.8 (red traces on the right). Panel (b) shows vertical slices, i.e. p_L as a function of Ω_0 at $\tilde{\psi}_w = .06$ (black traces on the left) and 0.15 (red traces on the right). Unlike the other two methods, the NN fairly captures the sharp increases in the GT profiles (solid trace).

IV. THE NEUTRAL CASE

A. The Solutions of the ODE's

In this section, we repeat the procedures of the previous section by applying them to a second case that is free of the locking bias. Many of the illustrations and much of the analysis of the previous sections are condensed to avoid repetition. We begin once again by integrating Eqs. (3)-(5) over the same range in time for the same 200×200 grid of control parameters, spanning $\tilde{\psi}_w = [0, 0.2]$ and $\Omega_0 = [0, 5]$. However, the locking bias previously introduced into the initial conditions is reduced in this case by narrowing the range in $\tilde{\psi}_t(0)$ to $[0, 2]$ and increasing the range in $\Omega_t(0)$ to $[0, 5]$. The range in $\theta_t(0)$ is kept the same as before: $[0, 2\pi]$

The phase diagram, i.e. time-asymptotic solutions of the ODE's as a function of $(\tilde{\psi}_w, \Omega_0)$ are shown in Fig. 12, with the color indicating the value of Ω_n , as in Fig. 3. Once again, the color blue represents the high frequencies ($\Omega_n \approx 1.0$) and the color red the low frequencies ($\Omega_n \approx 0.0$). A fea-

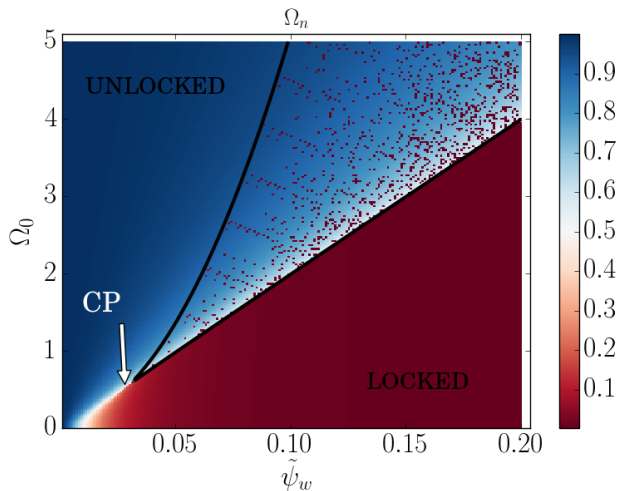


FIG. 12. The phase diagram: Time-asymptotic solutions of the ODE system in the control space $(\tilde{\psi}_w, \Omega_0)$ for the ‘neutral’ case, i.e. one that is less biased toward locking. The color represents the normalized rotation frequency $\Omega_n \equiv \Omega_t/\Omega_0$. The two black curves bound the hysteresis region that merge at the critical point (CP) of the system.

ture that immediately stands out compared to Fig. 3 of the previous section is the very low density of the locked states (red/orange pixels) residing in the hysteretic regime. In fact, only a small fraction ($\sim 9\%$) of solutions inside the hysteretic region lock. This is a consequence of the reduced locking bias applied to the initial conditions of the ODE’s.

The cumulative distribution functions (CDF’s) of the normalized order parameters are shown in Fig. 13. The most notable difference between these distribution functions and those shown in Fig. 2 is that the $\tilde{\psi}_n$ and Ω_n curves appear to have traded places, indicating that the majority of the solutions are now unlocked. The unlocked cluster also seems more diffuse, based on the distribution of Ω_n . The histograms (insets) also reflect these findings. The minima of the CDF’s suggest locking for $\Omega_n \lesssim 0.3$ or $\tilde{\psi}_n \gtrsim 0.5$, with the first criterion notably reduced compared to the threshold obtained from the CDF for the locking-biased case. Both criteria indicate approximately 42% of the total population to be locked, a significant reduction from the 55–56% observed for the locking-biased case.

B. Classification and Locking Probabilities

We start the process as before by classifying the solutions of the ODE’s, using the unsupervised classifier KMC. The figure for the KMC results is omitted here, as it shows approximately the same diagonal decision boundary as that shown in Fig. 6. The new centroid locations are $(\tilde{\psi}_n, \Omega_n) = (0.99, 0.01)$ and $(0.05, 0.91)$ for the locked and unlocked states, respectively, indicating a small shift in the location of the centroid for the unlocked cluster. This subtle shift is consistent with the increased population of unlocked solutions and their more diffuse distribution, indicated by Fig. 13. The

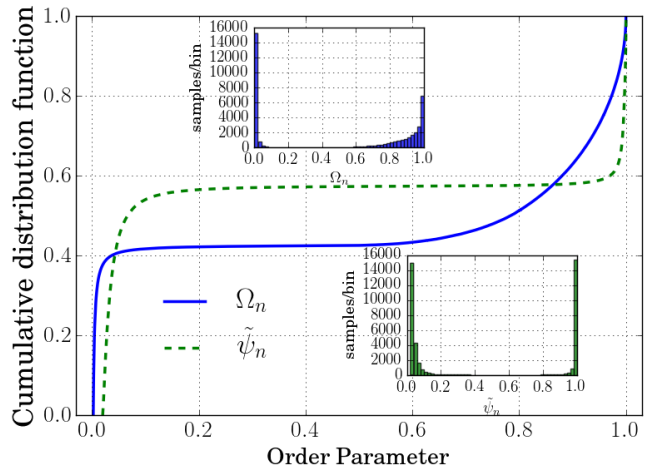


FIG. 13. The cumulative distribution functions (CDF’s) of the time-asymptotic normalized order parameters Ω_n and $\tilde{\psi}_n$ for the normal or unbiased case. The histograms in the inset show the number of samples per bin (estimating the probability density of the distribution) with respect to each order parameter.

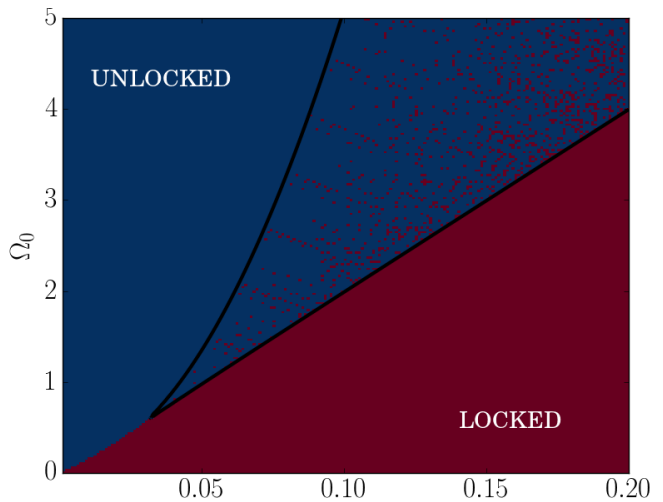


FIG. 14. The ‘speckling plot’, the intermediate processing step between unsupervised and supervised classifications: the KMC classification results are imposed on the phase diagrams for the neutral case—one of which appears in Fig. 12—whereby each point with $\tilde{\psi}_n > \Omega_n$ is marked locked (red) and each point with $\tilde{\psi}_n \leq \Omega_n$ is marked unlocked (blue). Clearly the density of locked points is lower.

tally for this case indicates that 57% of the population is unlocked and 43% is locked, a ratio that is approximately the inverse of the locking-biased case.

The same intermediate step that converts the phase diagrams, displaying color plots of the normalized order parameters in the control space, to a strict binary map is carried out here as well. The resulting speckling diagram, which is the counterpart of Fig. 7 appears in Fig. 14. Note the dramatic reduction in the number of locked (red) points that appear in the hysteretic region.

The ground truth conditional probability of locking $p_L^{(GT)}$ for the neutral case is shown in Fig. 15a. This probability is calculated by the same Monte Carlo procedure as that described in Section III C, except now the GT calculation uses the same range in the initial conditions as that used to produce Fig. 14. For clarity, only the probability in the hysteretic region is shown; $p_L^{(GT)} = 0$ holds above the top boundary segment and $p_L^{(GT)} = 1$ holds below the bottom segment, and are not plotted. Observe the much smaller values of $p_L^{(GT)}$ appearing in the color bar on the right, which is also indicated by the histogram of $p_L^{(GT)}$ values in the hysteretic region, shown in Fig. 15b. A gap in $p_L^{(GT)}$ appears here as well, similar to the one encountered in the case with the locking bias (Fig. 8b). However, this gap is now shifted toward lower probabilities. In fact, on the 200×200 grid, no point in the hysteretic region features a locking probability greater than $\sim 40\%$. A plot as in Fig. 11 (not shown) shows an extremely rapid increase in probabilities as the bottom boundary segment is approached, from around 40% to 100%.

The conditional probability of locking calculated by the neural network (NN) is shown in Fig. 16a for the less biased case under study here. The probability from the other two classifiers are omitted as the previous section has already established that NN yields the most accurate results. The neural network architecture has been altered from the previous configuration to produce once again the most accurate results with the smallest RMSE. To be specific, the network still consists of four hidden layers, but these four layers now contain 100, 100, 50, and 25 nodes, respectively. A ‘ReLU’ activation function is used again to propagate the information forward through the NN. Only the contours of p_L within the hysteretic region are shown in Fig. 16a. Note the $p_L = 0.5$ contour (dashed trace of Figs. 9a-c) does not appear here; in fact, on the 200×200 array, no pixel has an estimated probability greater than 40%.

The residual between the GT probability and the NN probability for the neutral case is shown Fig. 16b. The same filter as that applied to the residuals shown in Fig. 10 is used. The root-mean square error in the NN p_L for the area depicted in the figure is 0.058, approximately the same as the RMSE obtained from the NN applied to the case with the locking bias (Fig. 10c).

V. SUMMARY AND DISCUSSION

This paper presents a proof-of-principle of using MLC’s to forecast mode-locking by calculating the probability of a rotating tokamak plasma to lock to a static error field. It is demonstrated here that this locking probability can be applied successfully to sparse data, and provide meaningful and quantitative insight. This capability is relevant for realistic scenarios in which the amount of data that can be obtained from either experiments or high-fidelity simulations can be limited.

The data required for the MLC’s are generated by solving a simplified third order ODE system that describes the locking dynamics. The advantage of this model is the rapid

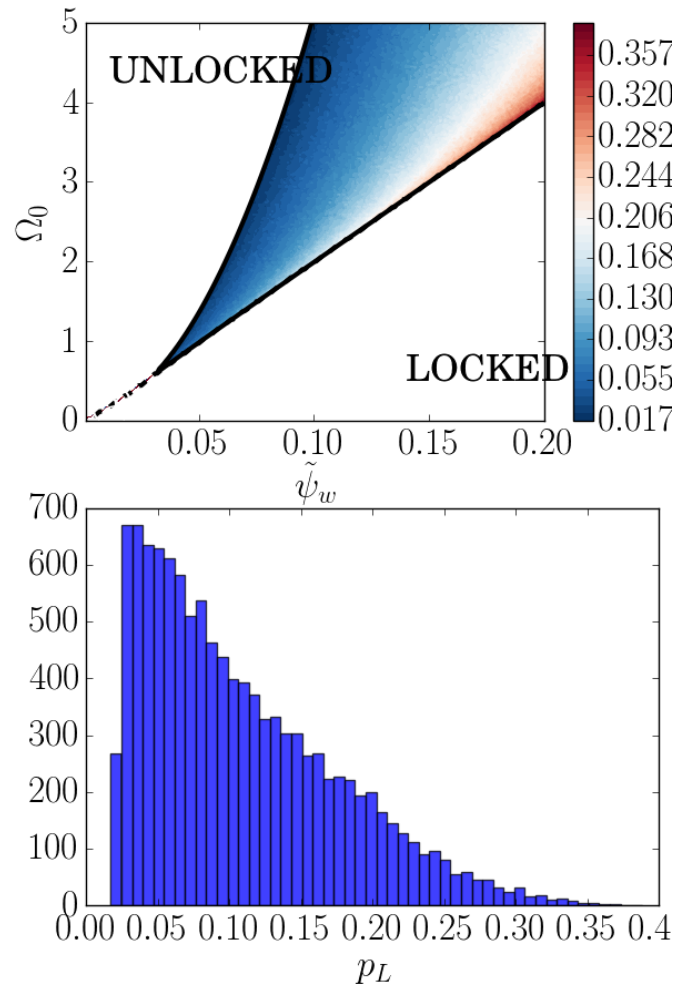


FIG. 15. (a) Ground truth conditional probability of locking $p_L^{(GT)}$ as calculated via the Monte Carlo approach of solving the ODE’s at each point in the control space for $N_i = 10^4$ randomly selected initial conditions, and using the diagonal decision boundary of Fig. 6 as the ultimate ground truth criterion for locking. (b) The histogram of $p_L^{(GT)}$ in the hysteretic region. The reduced locking bias is evident in both figures, as can be evidenced by the strong clustering of probabilities around 0.1, and very few values greater than $\sim 40\%$.

generation of data—a few minutes on a few dozen processors—required for the MLC’s, in addition to the well-demonstrated success of similar modeling in capturing the basic aspects of the locking dynamics that are useful for the MLC’s.

The locking probability is conditional on the two control parameters of the model, which are the error field magnitude ψ_w and the frequency Ω_0 associated with the momentum source. The ODE’s are integrated to the nonlinear steady-state for each point in a 200×200 uniform grid of these two control parameters. The dependent variables are the amplitude and phase of the tearing mode, $\tilde{\psi}_t$ and θ_t ; as well as Ω_t , the plasma rotation frequency at the tearing layer. The nonlinearly saturated values of these variables are the *order parameters* of the system. A single set of initial conditions for three order parameters are chosen randomly, over a prescribed range for each point in the (ψ_w, Ω_0) grid. The rela-

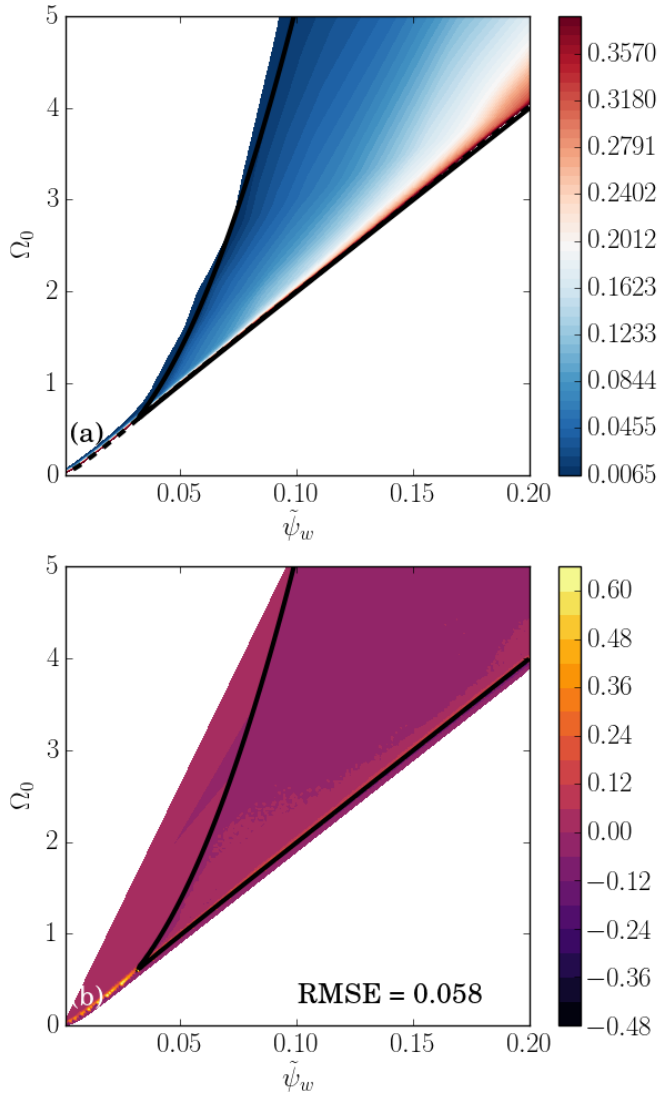


FIG. 16. (a) The contours of the conditional probability of locking for the neutral case (with a reduced locking bias), as obtained from a fully-connected feed-forward neural network (NN) with four hidden layers, each consisting of 100, 100, 50, and 25 nodes, respectively. (b) Residuals between the conditional probability of locking obtained from the Monte Carlo calculation and that obtained from the above mentioned NN.

tive dimensions of the set of initial condition from which a sample is taken randomly determine the tendency of the system to lock. Two scenarios are considered here: one with a strong locking bias and another with a reduced locking bias (neutral).

The time-asymptotic steady-state solutions of the ODE's agree with the analytic steady-state solutions, one of which emerges from the well-known torque balance equation, which is a cubic equation in Ω_t . The steady-state solutions show the presence of a *hysteretic* region in the control parameters $(\tilde{\psi}_w, \Omega_0)$, where the cubic equation of torque balance has three real roots. The upper and lower roots represent attractors, and initial conditions near the middle root converge

to one of the two attractor solutions. In other areas of the control parameter space, only one real root exists, corresponding to one steady state solution that is either locked or unlocked. The cubic equation of torque balance is analogous to the Van Der Waals equation of state that describes (classical) phase transitions between liquid and gaseous phases of matter.

The ODE solutions, i.e. the order parameters, are first normalized to span $[0, 1]$. Of these normalized parameters, $\tilde{\psi}_n$, θ_t , and Ω_n , the phase is redundant in the sense that it depends on $\tilde{\psi}_n$, independent of control parameters. The remaining two normalized order parameters show a strong clustering pattern, with the unlocked solutions accumulating around $(\tilde{\psi}_n, \Omega_n) \approx (0.0, 1.0)$ and locked solutions accumulating around $(\tilde{\psi}_n, \Omega_n) \approx (1.0, 0.0)$.

The hysteresis in the ODE system manifests itself in the control space $(\tilde{\psi}_w, \Omega_0)$ as a sizable region in which there is a strong mixing of locked and unlocked solution. This mixing implies a dependence on the randomly selected initial conditions, leading to an easily perceived density of locked states within this hysteretic region of control parameter space, as seen in Figs. 7, 14. This mixed population turns homogeneous near the bifurcation boundary segments: becoming unlocked near the segment of the boundary between the hysteretic region and the unlocked region, and becoming locked near the segment between the hysteresis and the locked region. The boundary segments of the hysteretic region merge at the critical point (CP) of the system below which smooth transitions are seen to occur between the two phases. This is analogous to behavior near the CP in the Van der Waals equation of state.

Because of the co-existence of the locked and unlocked states of the plasma in the hysteretic region, it is useful to describe the hysteretic behavior in terms of a probability of locking, conditional on the control parameters of the system. This probability is a measure of the relative size of the basin of attraction of the locked state. In the context of tokamak physics, this probability is a measure of the robustness of an unlocked state to a disturbance like a large sawtooth or a large ELM.

For calculating the locking probability via ML methods, we follow a two-fold approach. We first subject the normalized order parameters $(\tilde{\psi}_n, \Omega_n)$ of the model to an unsupervised classification scheme that labels each solution as locked or unlocked, free of human input. The *K-means clustering*²¹ (KMC) algorithm, which is a geometric method, is chosen for this task because of the approximately even density of solutions in the two clusters. The classification with KMC indicates locking for $\Omega_n < \tilde{\psi}_n$. Alternatively, a single order parameter can also be used as the input for KMC, which results in a locking criterion in terms of either Ω_n or $\tilde{\psi}_n$. These results have also been benchmarked with Gaussian mixture models (GMM), a probabilistic method that fits anisotropic Gaussians to the clusters. The different locking criteria from either KMC or GMM agree to better than 99% because of the very strong concentration of solutions around the two clusters. The normalization of the order parameters makes this clustering behavior evident, and this normalization leads to order parameters for which a Euclidean metric for the KMC classification is reasonable. Interestingly,

standard ML scaling methods have the option of scaling the data, but do not display this clustering behavior. These methods instead preserve the “L”-shaped distribution of the data, shown in Fig. 1a. Such scalings miss the most important aspect of our normalizations, the specific dependence upon the control parameters.

The class labels emerging from KMC provide the target values for the supervised classifiers used in the second stage of this work: training a series of supervised classifiers to calculate the conditional probability of locking $p(L|\tilde{\psi}_w, \Omega_0)$ (p_L for short) in the control space $(\tilde{\psi}_w, \Omega_0)$. We are primarily interested in the hysteretic region where $0 < p_L < 1$. The input space for the training consists of the control parameters of the system $(\tilde{\psi}_w, \Omega_0)$. The algorithms chosen for this task are support-vector machines with a radial basis function kernel (RBF SVM), a modified form of logistic regression (LR), and a fully-connected feed-forward neural network (NN). For the LR, we use a rational argument, a Padé approximant, for the argument of the sigmoid function to accommodate the distorted shape of the hysteretic region, in particular the merging of the boundary segments of the hysteretic region at the CP. The neural network contains four hidden layers of (200, 100, 100, 200) nodes, with each node using a linear rectified unit activation function to propagate the information forward through the network. To assess the accuracy of p_L calculated by the MLC’s, a ‘ground truth’ (GT) locking probability –unlikely to be available for more realistic scenarios–is also computed by a direct, but expensive, Monte Carlo method, where the ODE’s at each point in control space are integrated for $N_i = 10000$ different random initials conditions. The locking probability at each point is then given by the ratio of the tally of the locked points to N_i . In contrast to the GT calculation, which takes approximately a week of parallel computation on several dozen processors, the ML methods take about a minute to train. (Compared with the Monte Carlo computations described above, these methods use a *single* randomly selected set of initial conditions at each point in control parameter space.) These savings in computation time open up the possibility of applying these methods in real-time operations, once they are adequately trained on experimental data.

Of the three MLC’s that have been used, the NN produces probabilities that best agree with the GT probabilities, and result in the smallest residual and root-mean-square error (RMSE). The other two methods lead to results that do not match the ground truth probabilities as well, especially in capturing the finer features near the CP and along the boundary segments of the hysteretic region. They also yield a larger RMSE. The above network architecture is modified for the neutral case to produce the smallest RMSE.

Future work will include modifying the ODE’s to model the nonlinear saturation of the tearing mode²⁷ associated with the flattening of the plasma current, and by incorporating resistive wall (RW) effects. We have investigated the first effect partially: the saturation term changes the distribution of the solutions in the 2D space of the normalized order parameters $(\tilde{\psi}_n, \Omega_n)$, and thus leads to a new criterion for locking. The incorporation of the RW makes the system fifth order, as the RW is coupled both to the tearing layer and to

an outer perfectly-conducting boundary. In this case, it is the difference between the phase of tearing mode (TM) and the phase of the magnetic perturbation at the RW that determines the evolution of the TM and plasma rotation at the rational surface. Another fifth order system modeling the effect of the (2,1) tearing mode on the control system involving tokamak I and C-coils placed on either side of the (resistive) wall is presented in Ref. 28. It is exactly in situations like these where there are no analytic time-asymptotic solutions that the power of an unsupervised classifier truly emerges, for unambiguously distinguishing between locked and unlocked solutions.

The ODE model of the locking physics with the above modifications can be validated on the experimental data. The idea here is to tune the model’s parameters to the internal/external magnetic field and toroidal rotation measurements of DIII-D to determine if the NN can firstly predict the observed locking. If there is a lack of accuracy in the model, a separate NN can be incorporated into the model itself to capture the unknown physical effects that are absent in the physical model. Once the training is completed on a series of shots, the NN can then be used to produce charts of the locking probability in real-time operations. If the locking probability in a certain region of the control space exceeds the operator’s tolerance, active feedback can be engaged to prevent locking.

ACKNOWLEDGEMENTS

This work was supported by the DOE Office of Science collaborative Grant Nos. DE-SC0019016 and DE-SC0014005, respectively.

Appendix A: Analogy with the Van der Walls equation of state

It is interesting to note that the Van der Walls (VDW) equation of state (EOS) is also cubic as a function of the density n of a non-ideal gas:

$$abn^3 - an^2 + (T + bp)n - p = 0, \quad (\text{A1})$$

where T and p are the temperature and pressure of the material; a and b are the VDW constants related to the intermolecular potential and the molecular size, which have different values for each gas; and R is the universal gas constant.

Defining the normalized frequency $x \equiv \Omega_t/\Omega_0$ for Eq. (9) with the following simplifications $\mu = \Delta_1 = 1$, as well as setting $a = b = R = 1$ in Eq. (A1), shows that these two equations are analogous:

$$x^3 - x^2 + \left(\frac{\tilde{\psi}_w^2}{\Omega_0^2} + \frac{1}{\Omega_0^2} \right) x - \frac{1}{\Omega_0^2} = 0, \quad (\text{A2})$$

$$n^3 - n^2 + (RT + p)n - p = 0. \quad (\text{A3})$$

These equations establish an equivalence between the VDW control parameters (p, T) and ours $(\Omega_0, \tilde{\psi}_w)$, with the following analogs: $n \leftrightarrow x$ and, in units with $R = 1$,

$$p \leftrightarrow 1/\Omega_0^2, \quad T \leftrightarrow \tilde{\psi}_w^2/\Omega_0^2; \quad (\text{A4})$$

$$\Omega_0 \leftrightarrow 1/\sqrt{p}, \quad \tilde{\psi}_w \leftrightarrow \sqrt{T/p}. \quad (\text{A5})$$

The locked and unlocked states in this paper correspond, respectively, to the gaseous and liquid phases in the VDW equation of state.

One difference between the locking-unlocking bifurcation problem and the phase transformation problem exists in the hysteretic regime. In the former there is a probability of locking $p(L|\tilde{\psi}_w, \Omega_0)$, which varies from zero at the boundary segment with the unlocked region (the upper black curve in Fig. 3) to unity at the segment with the locked region (the lower black curve there), measuring the respective number of initial conditions leading to a locked state. In the phase transformation case, the gaseous and liquid phases can coexist, with the fraction of each roughly corresponding roughly to $p(L|\tilde{\psi}_w, \Omega_0)$. The analogy would possibly be more complete if the ODE for the locking problem included additive noise that can cause a transition from locked to unlocked and back for control parameters in the hysteretic region, similar to the effect of fluctuations on phase transformations. With this noise another measure of the probability of locking might be the amount of time spent, in the presence of the noise, near the locked state.

The VDW critical point, with large values of the control parameters (p, T) , corresponds to small values of $(\tilde{\psi}_w, \Omega_0)$ (c.f. Eqs. (A4) and (A5)). The possibility of traversing a path in the control space (p, T) above the critical point, with the state smoothly transforming from the gaseous state to the liquid state, is the exact analog to tracing a path in Fig. 1 below the critical point.

Appendix B: Description of the machine learning classification (MLC's)

In this appendix we outline the ML methods we employ in our two-step scheme. The first step is to do clustering, in order to label states in the normalized order parameter space as locked or unlocked. In the second step we use this classification to estimate the probability of locking as a function of the two control parameters, especially in the hysteretic region, where both locked and unlocked states can co-exist.

1. Unsupervised Classifiers

We use *K-means clustering*^{21,22} to classify each solution of the ODE's as locked or unlocked, without human input. The aim of KMC is to partition N sample points into K clusters in which each sample belongs to the cluster with the nearest mean cluster center or centroid. In our application, we use this classification scheme with $K = 2$ to classify all of the points in the normalized order parameter space as locked or

unlocked states. This technique is used in image segmentation and compression. One begins by randomly choosing K points as the initial estimates of the centroids of K clusters. One then assigns each sample to the cluster with the closest centroid, and then re-computes K new centroids by using the newly re-labeled samples in each cluster. The two phases of re-assigning samples to the clusters and re-computing the cluster centroids continue until the centroid locations no longer change or, equivalently, there is no further change in assignments. The algorithm minimizes the sum of the squares of the distances of each data point to the closest centroid. It also has the capability of predicting the class of any new or additional samples by associating the new sample with the cluster with the nearest centroid. Since this method depends on distances between points, the choice of the metric is important and, as we have discussed, choosing the Euclidean metric on the space of normalized parameters $\tilde{\psi}_n$ and Ω_n is a natural choice. The labels or categories of this classification form the *target values* required in the training of the supervised classifiers that are used to calculate locking probabilities as a function of the two control parameters.

We have also applied Gaussian-mixture models (GMM) to our data. GMM is another unsupervised classification method which fits K —many anisotropic Gaussians to the data, where each Gaussian represents one cluster. GMM makes ‘soft’ assignments to each data point as opposed to the ‘hard’ assignments made by KMC.

2. Supervised Classifiers

We make use of three classes of supervised classifiers to compute the locking probability as a function of the two control parameters. These are support-vector machines (SVM), logistic regression (LR), and a fully-connected feed-forward neural network (NN), also known as a multi-layer perceptron (MLP). These methods—as do all supervised classifiers—require the training set to contain the targets as well as the inputs. The targets for the training are the binary classification of the solutions into locked and unlocked states, supplied in this case by KMC, as described above and implemented in Section III B. The paragraphs below contain descriptions of each of the three supervised classifiers used in this work. A comprehensive review of these classifiers can be found in Ref. 21.

To ease the reader into the following MLC descriptions, we first define some terms and notation. In the presence of multiple features like the height and weight of a person, the input vector samples take the form of vectors lined up to form a 2D *design matrix* $\mathbf{X} \equiv X_{\mu n}$. Here, $n = 1, \dots, N$ tracks the samples of a data set with N samples and $\mu = 1, \dots, D$ tracks the components of the *input or feature space*. For our application, the inputs are the coordinates $(\tilde{\psi}_w, \Omega_0)$ of the 200×200 grid in the control space, yielding $D = 2$ and $N = 40000$. Thus, $\mathbf{X} = (\mathbf{x}_1, \mathbf{x}_2) = \{\mathbf{x}_\mu\}$ where the *bold* symbol is to remind the reader that each feature \mathbf{x}_μ is a vector containing all of the N samples. For simplicity the subscript n is suppressed in most of these definitions. Lastly, the notation $\mathbf{x} = (x_1, x_2) = \{x_\mu\}$ is reserved for a single sample

and similarly, \mathbf{x}_n for the n^{th} sample.

Support vector machine: A SVM^{21,29} separates two classes by maximizing the margin, defined as the smallest distance between the linear decision boundary separating the two classes and any of the samples belonging to either class. Maximizing this margin leads to a particular choice of decision boundary, whose location is determined by a subset of the closest data points, known as support vectors. One can obtain a nonlinear boundary by performing the *kernel trick* or *kernel substitution*³⁰. A common choice for a Kernel FOR SVM are radial basis functions (RBF), which produce an infinite dimensional feature space. The kernel trick is equivalent to constructing a higher dimensional nonlinear feature space, which augments \mathbf{X} to a higher dimensional nonlinear feature space $\phi(\mathbf{X})$, e.g. $\mathbf{X} = (\mathbf{x}_1, \mathbf{x}_2) \rightarrow \phi(\mathbf{X}) = (1, \mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_1^2, \mathbf{x}_1\mathbf{x}_2, \mathbf{x}_2^2)$. A linear classification in this feature space leads to a nonlinear classification in the original space. Note that SVM must account for overlapping class distributions in this case because of the strong intermixing of the locked and unlocked states in the hysteretic region, as indicated by Figs. 7 and 14. This is accomplished by penalizing the points on the wrong side of the boundary with a penalty that increases with the distance from the boundary. This modification yields a *soft-margin* classifier.

A support-vector machine does not provide probabilistic outputs. However, probabilities out of an SVM classification can be obtained by fitting the SVM output to a logistic sigmoid function, which is described in the following paragraph. This is known as the Platt's method³¹.

Logistic Regression: LR provides a probabilistic classifier by best fitting a logistic sigmoid function $\sigma(x) = 1/(1 + e^{-x})$ to a binary classification problem. LR is a maximum likelihood estimator of the parameters of the logistic sigmoid, and in its most primitive form, provides an estimate of the probability of belonging in either one of two classes: locked and unlocked states of the plasma in this case.

For the present application and focusing on a single sample \mathbf{x} with features (x_1, x_2) and bias³² $x_0 = 1$, the sigmoid function $\sigma(\mathbf{w} \cdot \mathbf{x})$ corresponds to the probability of locking $p(L|\mathbf{w}, \mathbf{x})$ conditional on the weights $\mathbf{w} = (w_0, w_1, w_2)$. Here the inner product is performed over the features: $\mathbf{w} \cdot \mathbf{x} = w_0x_0 + w_1x_1 + w_2x_2$. The conditional probability of being unlocked is $p(U|\mathbf{w}, \mathbf{x}) = 1 - \sigma(\mathbf{w} \cdot \mathbf{x})$. For the n^{th} sample of the data set $\{\mathbf{x}_n, t_n\}$, we define $t_n = 1$ for locked states and $t_n = 0$ for unlocked states, so $p(t_n = 1|\mathbf{w}, \mathbf{x}_n) = \sigma(\mathbf{w} \cdot \mathbf{x}_n) \equiv \sigma_n$ and $p(t_n = 0|\mathbf{w}, \mathbf{x}_n) = 1 - \sigma_n$. Then the likelihood for locking on a single trial can be written as $p(t_n|\mathbf{w}, \mathbf{x}_n) = \sigma_n^{t_n} (1 - \sigma_n)^{1-t_n}$. Assuming independence of all the measurements for all values t_n , the likelihood to be maximized is

$$p(\mathbf{t}|\mathbf{w}) = \prod_{n=1}^N \sigma_n^{t_n} (1 - \sigma_n)^{1-t_n}. \quad (\text{B1})$$

(Interpreting this in a Bayesian sense, the posterior is – except for normalization – equal to the likelihood, assuming that the prior is taken to be uniform.) The negative log-

likelihood is the *the cross entropy*²¹

$$E(\mathbf{w}) = -\ln p(\mathbf{t}|\mathbf{w}) = -\sum_{n=1}^N [t_n \ln \sigma_n + (1-t_n) \ln(1-\sigma_n)], \quad (\text{B2})$$

The weights found by the process of minimizing $E(\mathbf{w})$ for the data points (\mathbf{x}_n, t_n) give an estimate for the probability of belonging to the locked class.

For a 2D input space, LR produces probability contours that are parallel straight lines, which do not conform to the shape of the hysteretic region in Figure 3. To capture the properties of the hysteretic region, the linear bases of LR are augmented to form a nonlinear feature space, as discussed above for SVM. This generalization, based on a linear combination of the feature vectors, is known as a *generalized linear model* (GLM), in that the argument of the sigmoid is linear in the parameters, i.e. the weights \mathbf{w} . What is carried out in Section III C is a further generalization of this; we take as the argument of the sigmoid a rational function of the form $Q(\mathbf{x})/L(\mathbf{x})$, a ratio of a quadratic to a linear polynomial. This *rational logistic regression* (RLR) is no longer a GLM, as the argument of the sigmoid is not linear in \mathbf{w} . In fact, it results in a nonlinear optimization process—of 8 weights and biases in this case (6 for the quadratic polynomial and three for the linear, but the bias in the denominator can be scaled to unity) –that is extremely sensitive to the initial values of the weights, and for which ScikitLearn's LR algorithm is ill suited. Thus, we carry out the RLR by calculating the log likelihood function explicitly and then minimizing it with respect to \mathbf{w} , using various Python optimization routines. The advantage of the rational formulation is its potential to deal with the tangency $\delta\Omega_0 \propto (\delta\tilde{\psi}_w)^{3/2}$ between the two bifurcation boundary segments that meet at the CP. This use of a rational function (or Padé approximant^{33,34}) in the argument of the logistic sigmoid deserves more study, but this is outside the scope of this paper.

Multi-layer perceptron or fully-connected artificial neural network (NN): This is an early class of feed-forward artificial neural networks. It consists of at least three layers of nodes: an input layer, one or more hidden layers, and an output layer. The information is propagated forward through the layers by the use of nonlinear activation function like the sigmoid $\sigma(x)$ defined above or rectified linear units (ReLU's)³⁵ $\tau(x) = \max(x, 0)$. These activation functions model the firing of biological neurons. Fully connected means that each node in one layer is connected to every node in the preceding or the following layer through the activation functions. The overall network function, for a single hidden layer with M nodes (neurons) and D -many inputs forming the input vector $\mathbf{x} = (x_1, x_2, \dots, x_D) \equiv \{x_\mu\}$, again for a single sample, takes the following form for a single output:

$$y(\mathbf{x}, \mathbf{w}) = f \left(\sum_{j=1}^M w_j^{(2)} f \left(\sum_{\mu=1}^D w_{j\mu}^{(1)} x_\mu + \underbrace{w_{j0}^{(1)}}_{\text{biases}} \right) + \underbrace{w_0^{(2)}}_{\text{biases}} \right), \quad (\text{B3})$$

where $D = 2$ for the present application and the activation function f is chosen to be a ReLU for reasons dis-

cussed in Section III C. The biases can be grouped together with the weights as in the last section, writing, for example: $\sum_{\mu=1}^D w_{j\mu}^{(1)} x_{\mu} + w_{j0}^{(1)} = \sum_{\mu=0}^D w_{j\mu}^{(1)} x_{\mu}$. Note that the weight vectors \mathbf{w} are now 2D arrays $w_{qr}^{(i)}$, with their superscript denoting the order of mapping between the layers: e.g. superscript (1) corresponds to the nonlinear mapping from the inputs to the first hidden layer in Eq. (B3). For the single-layer example of Eq. (B3), the number of parameters that the NN must optimize is $D \times (M + 1) + (M + 1)$. For the network architecture employed in Section III C, which has four hidden layers that contain 200, 100, 100, 200 nodes, respectively; the number of weights are approximately $(2 \times 200) + (200 \times 100) + (100 \times 100) + (100 \times 200) + 200 = 5.6 \times 10^4$.

For the present application, each output of the NN is exactly the conditional probability of belonging to the locked state, $p(L|\tilde{\psi}_w, \Omega_0)$. This nonlinear propagation of information from the inputs to the outputs within an NN bears some resemblance to LR. However, the network's structure of passing the inputs through a series of layers allows for far greater flexibility due to a much larger number of tunable parameters (weights and biases) than LR. Whereas each input is directly (and nonlinearly) mapped to an output in LR, in a NN the contribution from all of the inputs is nonlinearly mapped to each node in all of the hidden layers successively and then eventually to the output. This highly flexible architecture of a NN allows it to approximate any continuous function on a compact input domain to arbitrary accuracy, provided the network has a sufficiently large number of hidden units. This *universal approximation* property of NN's has its origins in the works by Kolmogorov³⁶ and Arnold³⁷, and was fully brought to light by Funahashi³⁸, Cybenko³⁹, Hornik^{40,41}, and a few others.

Learning occurs in the neural network by changing connection weights w_1, \dots, w_N and biases w_0 for each layer after each piece of data is processed, to minimize the amount of error in the output compared to the target t_k values (expected result). The error (cost) function is defined as:

$$E(\mathbf{w}) = \frac{1}{2} \sum_{n=1}^N |y_n(\mathbf{x}, \mathbf{w}) - t_n|^2 \quad (\text{B4})$$

for a N -many outputs.

The methods used to minimize $E(\mathbf{w})$ are variants of the gradient descent (GD) method²¹: each iteration moves the weights \mathbf{w} in the direction opposite to the gradient of $E(\mathbf{w})$. The corrections to each weight is backpropagated from the outputs to the inputs for each iteration⁴². Other enhancements such as stochastic (online) gradient descent⁴³ are often used, because of the inefficiency of the standard GD, where parameter updates are performed on the basis of a single sample or subset of samples that is picked randomly at each iteration.

We use Python's *Scikit-learn* libraries to apply the said ML algorithms.

¹R. Fitzpatrick and T. C. Hender. *Physics of Fluids B: Plasma Physics* (1989-1993), 3(3):644–673, 1991.

²R Fitzpatrick. Interaction of tearing modes with external structures in cylindrical geometry (plasma). *Nuclear Fusion*, 33(7):1049, 1993.

- ³R. J. Buttery, R. J. La Haye, P. Gohil, G. L. Jackson, H. Reimerdes, and E. J. Strait. The influence of rotation on the betan threshold for the 2/1 neoclassical tearing mode in diiii-d. *Physics of Plasmas*, 15(5):056115, 2008.
- ⁴F. D. Halpern, A. H. Kritz, G. Bateman, A. Y. Pankin, R. V. Budny, and D. C. McCune. Predictive simulations of iter including neutral beam driven toroidal rotation. *Physics of Plasmas*, 15(6):062505, 2008.
- ⁵C. G. Windsor, G. Pautasso, C. Tichmann, R. J. Buttery, T. C. Hender, JET EFDA Contributors, and the ASDEX Upgrade Team. A cross-tokamak neural network disruption predictor for the JET and ASDEX upgrade tokamaks. *Nuclear Fusion*, 45(5):337–350, apr 2005.
- ⁶A. Murari, J. Vega, D. Mazon, G.A. Rattá, M.Gelfusa, A. Debie, C. Boulbe, and B. Faugeras. New signal processing methods and information technologies for the real time control of jet reactor relevant plasmas. *Fusion Engineering and Design*, 86(6):544–547, 2011. Proceedings of the 26th Symposium of Fusion Technology (SOFT-26).
- ⁷B. Cannas, A. Fanni, P. Sonato, M. K. Zedda, and JET EFDA Contributors. A prediction tool for real-time application in the disruption protection system at JET. *Nuclear Fusion*, 47(11):1559–1569, oct 2007.
- ⁸J. Kates-Harbeck, A. Svyatkovskiy, and W. Tang. Predicting disruptive instabilities in controlled fusion plasmas through deep learning. *Nature*, 568(7753):526–531, 2019.
- ⁹Y. Fu, D. Eldon, K. Erickson, K. Kleijwegt, L. Lupin-Jimenez, M. D. Boyer, N. Eidietis, N. Barbour, O. Izacard, and E. Kolemen. Machine learning control for disruption and tearing mode avoidance. *Physics of Plasmas*, 27(2):022501, 2020.
- ¹⁰R. Fitzpatrick. Bifurcated states of a rotating tokamak plasma in the presence of a static error-field. *Physics of Plasmas*, 5(9):3325–3341, 1998.
- ¹¹Y. Liu, J. W. Connor, S. C. Cowley, C. J. Ham, R. J. Hastie, and T. C. Hender. Toroidal curvature induced screening of external fields by a resistive plasma response. *Physics of Plasmas*, 19(7):072509, 2012.
- ¹²J. M. Finn, A. J. Cole, and D. P. Brennan. Real frequency tearing layers with parallel dynamics and the effect on error field locking and resistive wall modes. *Physics of Plasmas*, 26(10):102505, 2019.
- ¹³A. J. Cole, J. M. Finn, C. C. Hegna, and P. W. Terry. Forces and moments within layers of driven tearing modes with sheared rotation. *Physics of Plasmas*, 22(10):102514, 2015.
- ¹⁴R. Fitzpatrick. Linear and nonlinear response of a rotating tokamak plasma to a resonant error-field. *Physics of Plasmas*, 21(9):092513, 2014.
- ¹⁵C. Akçay, J. M. Finn, A. J. Cole, and D. P. Brennan. Nonlinear error field response in the presence of plasma rotation and real frequencies due to favorable curvature. *Physics of Plasmas*, 27(3):032302, 2020.
- ¹⁶J. M. Finn, A. J. Cole, and D. P. Brennan. Error field penetration and locking to the backward propagating wave. *Physics of Plasmas*, 22(12):120701, 2015.
- ¹⁷Because of the presence of $\tilde{\psi}_t$ in Eq. (5), we have compared with integrating the real and imaginary components of ψ_c , finding comparable results: the possible jumps $\pm 2\pi$ in θ_t near $\psi_t = 0$ have little effect on the overall results.
- ¹⁸R. J. La Haye, D. P. Brennan, R. J. Buttery, and S. P. Gerhardt. Islands in the stream: The effect of plasma flow on tearing stability. *Physics of Plasmas*, 17(5):056110, 2010.
- ¹⁹D Chandra, A Sen, P Kaw, M.P Bora, and S Kruger. Effect of sheared flows on classical and neoclassical tearing modes. *Nuclear Fusion*, 45(6):524–530, may 2005.
- ²⁰An alternate choice $\tilde{\psi}_{n,2} = \tilde{\psi}_t / \tilde{\psi}_w$ gives points on the rectangle $[0, 10] \times [0, 1]$ with $|\Delta_1| = 0.1$, and a Euclidean metric on this space overemphasizes differences in $\tilde{\psi}_t$ relative to those in Ω_t .
- ²¹C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.
- ²²J. MacQueen. Some methods for classification and analysis of multivariate observations. In *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, volume 1, pages 281–297. Oakland, CA, USA, 1967.
- ²³This calculation requires integrating the ODE system a total of 4×10^8 times.
- ²⁴C. J. Feng, A. C. Gowrisankar, A. E. Smith, and Z. S. Yu. Practical guidelines for developing bp neural network models of measurement uncertainty data. *Journal of Manufacturing Systems*, 25(4):239–250, 2006.
- ²⁵A. Choromanska, M. Henaff, M. Mathieu, B. A. Gérard, and Y. LeCun. The loss surfaces of multilayer networks. 2015.

- ²⁶A. H. Glasser, C. R. Sovinec, R. A. Nebel, T. A. Gianakon, S. J. Plimpton, M. S. Chu, D. D. Schnack, and the NIMROD team. The NIMROD code: A new approach to numerical plasma physics. *Plasma Phys. Control. Fusion*, 41:A747, 1999.
- ²⁷P. H. Rutherford. Nonlinear growth of the tearing mode. *The Physics of Fluids*, 16(11):1903–1908, 1973.
- ²⁸K E J Olofsson, W Choi, D A Humphreys, R J La Haye, D Shiraki, R Sweeney, F A Volpe, and A S Welander. Electromechanical modelling and design for phase control of locked modes in the DIII-d tokamak. *Plasma Physics and Controlled Fusion*, 58(4):045008, feb 2016.
- ²⁹B. E. Boser, I. M. Guyon, and V. N. Vapnik. A training algorithm for optimal margin classifiers. In *Proceedings of the Fifth Annual Workshop on Computational Learning Theory, COLT '92*, page 144–152, New York, NY, USA, 1992. Association for Computing Machinery.
- ³⁰K. R. Muller, S. Mika, G. Ratsch, K. Tsuda, and B. Scholkopf. An introduction to kernel-based learning algorithms. *IEEE transactions on neural networks*, 12(2):181–201, 2001.
- ³¹J. Platt. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in large margin classifiers*, 10(3):61–74, 1999.
- ³²This is not to be confused with the locking bias that controls the ODE system’s tendency toward locking.
- ³³G. A. Baker and P. Graves-Morris. *Padé Approximants*. Cambridge University Press, 2009.
- ³⁴A. J. Cole and J. M. Finn. Variational principles with Padé approximants for tearing mode analysis. *Physics of Plasmas*, 21(3):032508, March 2014.
- ³⁵V. Nair and G. E. Hinton. Rectified linear units improve restricted Boltzmann machines. In *ICML*, 2010.
- ³⁶A. Kolmogorov. On the representation of continuous functions of many variables by superposition of continuous functions of one variable and addition. *Proceedings of the USSR Academy of Sciences*, (108):179 – 182, 1956.
- ³⁷V. I. Arnold. On the representation of continuous functions of three variables as superpositions of continuous functions of two variables. *Proceedings of the USSR Academy of Sciences*, (114):679 – 681, 1957.
- ³⁸K. I. Funahashi. On the approximate realization of continuous mappings by neural networks. *Neural Networks*, 2(3):183 – 192, 1989.
- ³⁹G. Cybenko. Approximation by superpositions of a sigmoidal function. *Mathematics of control, signals and systems*, 2(4):303–314, 1989.
- ⁴⁰K. Hornik, M. Stinchcombe, and H. White. Multilayer feedforward networks are universal approximators. *Neural networks*, 2(5):359–366, 1989.
- ⁴¹K. Hornik. Approximation capabilities of multilayer feedforward networks. *Neural Networks*, 4(2):251 – 257, 1991.
- ⁴²D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Learning representations by back-propagating errors. *nature*, 323(6088):533–536, 1986.
- ⁴³Y. LeCun, L. Bottou, G. Orr, and K. Muller. Efficient backprop. In G. Orr and Muller K., editors, *Neural Networks: Tricks of the trade*. Springer, 1998.