Journal of Intelligent Manufacturing
https://doi.org/10.1007/s10845-023-02291-1

Using GANs to predict milling stability from limited data

Shahrbanoo Rezaei[1] · Aaron Cornelius[2] · Jaydeep Karandikar[3] · Tony Schmitz[2,3] · Anahita Khojandi[1]

Received: 24 April 2023 / Accepted: 25 November 2023
© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2024

Abstract

Milling is a key manufacturing process that requires the selection of operating parameters that provide efficient performance. However, the presence of chatter, a self-excited vibration causing poor surface finish and potential damage to the machine and cutting tool, makes it challenging to select the appropriate parameters. To predict chatter, stability maps are commonly used, but their generation requires expensive data, making it difficult to employ these maps in industry. Therefore, there is a pressing need for an approach that can accurately predict stability maps using limited experimental data. This study introduces the new Encoder GAN (EGAN) approach based on Generative Adversarial Networks (GANs) that predicts stability maps using limited experimental data. The approach consists of the encoder, generator, and discriminator subnetworks and uses the trained encoder and generator to predict the target stability map. This versatile method can be applied to various tool setups and can accurately predict stability maps with limited experimental data (five to 10 cutting tests) even when there is little information available for unknown parameters. The study evaluates the proposed approach using both numerical data and experiments and demonstrates its superior performance compared to state-of-the-art benchmarks.

Keywords Milling · Chatter · Generative adversarial network · Deep learning

Introduction

Milling is a common manufacturing process which uses a rotating cutting tool to remove material from a workpiece (Yan et al., 2023). Since milling is expensive, it is important to select productive operating parameters, including spindle speed n, radial depth of cut a, and axial depth of cut b, which enable efficient removal of material from the workpiece (Fig.1) (Schmitz & Smith, 2019).

Ideally, these values should be set as high as possible to minimizecuttingtime.However,therearevariousconstraints thatpreventsomeparametercombinationsfrombeingviable, such as chatter. Chatter is a self-excited vibration that occurs atcertainselectionsofmachiningparametersandcausespoor surface finish and can potentially damage the cutting tool and machine (see Fig.2c) (Unver & Sener, 2023; Jauhari et al., 2023). If the tool tip and workpiece dynamics are notconsideredwhenselectingmachiningparameters,chatter may occur, where varying chip thickness causes regenerative vibration.

Stable (i.e., chatter-free) machining parameters can be selected using the stability map (Fig.2a), which separates spindle speed-axial depth pairs that are stable from those that exhibit chatter. The stability map can be predicted based on the tool tip frequency response function (FRF) and the toolworkpiece cutting force model (Deng et al., 2023). However, in practice, these are not typically known and the

This manuscript has been authored in part by UT-Battelle, LLC, under contract DE-AC05-00OR22725 with the US Department of Energy (DOE). The US government retains and the publisher, by accepting the article for publication, acknowledges that the US government retains a nonexclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this manuscript, or allow others to do so, for US government purposes. DOE will provide public access to these results of federally sponsored research in accordance with the DOE Public Access Plan (http://energy.gov/downloads/doepublic-access-plan).

B Anahita Khojandi khojandi@utk.edu

[1] Industrial & System Engineering, University of Tennessee, Knoxville, TN, USA

[2] Mechanical, Aerospace, and Biomedical Engineering, University of Tennessee, Knoxville, TN, USA

[3] Manufacturing Science Division, Oak Ridge National Lab, Oak Ridge, TN, USA

Published online: 13 January 2024

stability map and optimal machining parameters must be inferred from experimental test cuts.

The importance of considering the system's vibration response when selecting machining parameters has been thoroughly established in the literature. Despite numerous research efforts to improve the accuracy of stability maps, the theoretical and experimental results sometimes differ, which can require validation testing prior to final parameter selection. The dynamic model's inputs, the machining parameters,andthesolutionalgorithmallaffectstabilitymap accuracy(Chenetal.,2021).Modeladvancesincludeprocess damping (Wan et al., 2017) and symmetry breaking (Totis et al., 2019). In addition, to verify the accuracy of the solving algorithm, several methods such as full-discretization method (FDM) (Ding et al., 2010), zeroth-order approximation (Budak & Altintas, 1998), and semi-discretization method (SDM) (Insperger & Stépàn, 2002) have been introduced. Despite these improvements, a key challenge that remains is knowledge of the tool tip dynamics and the cutting force coefficients (Chen et al., 2021).

Currently, modal impact testing is the technique that is mostfrequentlyusedtomeasurethetooltipdynamics.Inthis method, the tool tip is excited using an instrumented hammer and the vibration response is measured using a transducer (commonly a piezoelectric accelerometer). While the measurementtechniqueiswell-understood,errorsourcesremain, which include the influence of the load condition (Postel et al., 2018), the mass loading effect of the accelerometer (Özs¸ahin et al., 2010), and the centrifugal forces, gyroscopic moments, and temperature increase in cutting operations (Gupta et al., 2020). Similarly, the estimate of the cutting force coefficients is also subject to various errors and uncertainties. These include fitting errors in regression methods and coefficient variation with cutting conditions (Dang et al., 2010; Grossi et al., 2015; Campatelli & Scippa, 2012).

Due to these inherence uncertainties in the tool tip dynamicsandcuttingforcecoefficients,uncertaintiesareintrod uced into the deterministic stability maps. To address this limita-

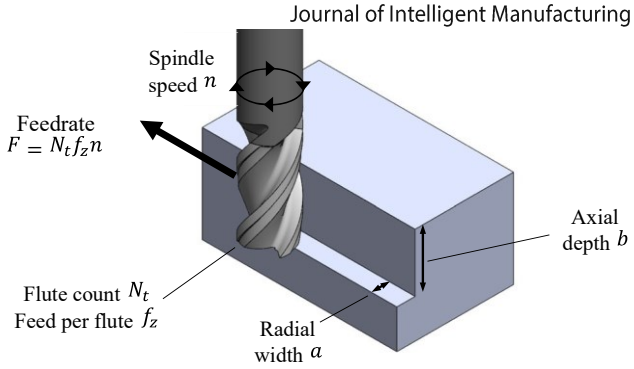
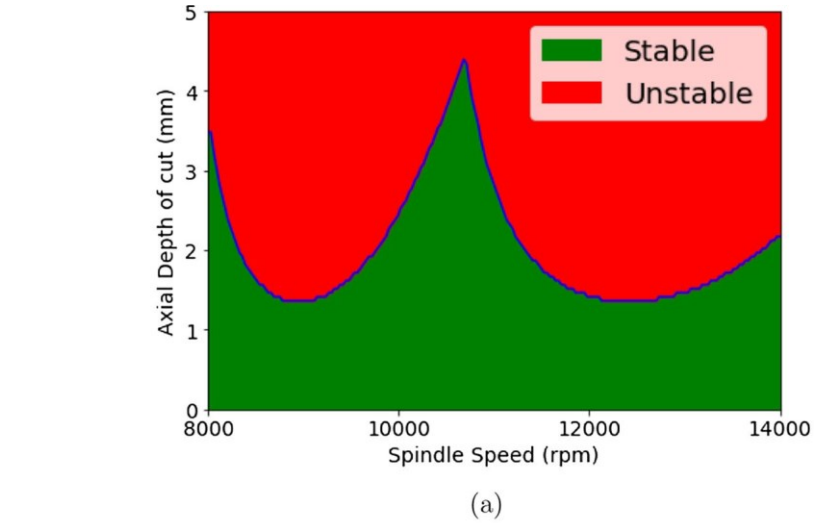


Fig. 1 Illustration of the milling operation. N_t represents the number of teeth (or flutes) on the milling cutter. f_z is the feed per flute. It indicates the chip thickness that each tooth of the milling cutter removes from the material as it passes through it

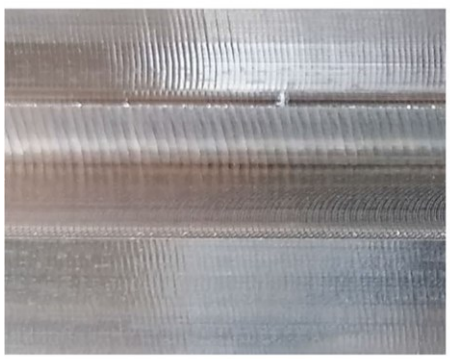
tion researchers have focused on approaches that identify dynamics during machining. Operational modal (Zaghbani & Songmene, 2009) analysis has been implemented to assess the machine's response using an accelerometer during routine machining operations. Although the identification of the damping ratio and natural frequency is straightforward, the dynamic stiffness is not easily obtained using this approach. In Özs¸ahin et al. (2015), an inverse identification approach is proposed in which chatter frequencies and limiting axial depthsofcutaredeterminedexperimentally.Inthisapproach, data for two spindle speeds that are marginally different are needed for the inverse identification of the tool tip dynamics. Similar approaches are proposed in Eynian (2019), Grossi (2016). However, these approaches necessitate specialized testsunderpredeterminedconditions,whichmakethemchallen ging to implement in shop floor environments.

Recently,machinelearninghasbeenappliedtomachining stability modeling. Karandikar et al. (2020) propose a novel Bayesian learning approach to predict the stability limit and find the optimal parameters utilizing test data only without knowledgeoftheunderlyingcuttingforcecoefficientsortool tip dynamics. Simulation results from physics-based models offer another data source. In this approach, physics-based models are used to generate stability maps. The predicted stability maps are then discretized and used as training data in machine learning approaches, such as artificial neural networks (Cherukuri et al., 2019; Oleaga et al., 2018; Yin et al., 2023), K-nearest neighbors (Greis et al., 2020), and support vector machines (Friedrich et al., 2017). For instance, Cornelius et al. (2021) propose a physics-guided Bayesian approach in which first, prior uncertainties of parameters are considered to obtain the probabilistic stability map using the physics-based stability model. Second, this probabilistic stability map is updated in each

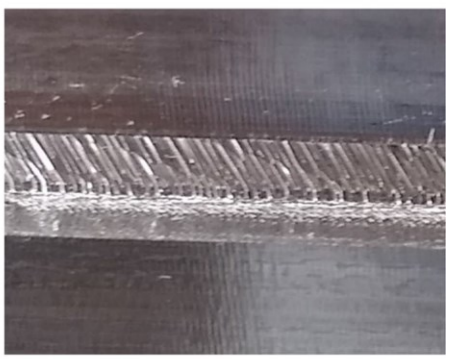
iteration based on the new information obtained from the cutting test. A similar Bayesian approach is proposed in Chen et al. (2021). Schmitz et al. (2022) presents a novel approach to improving milling stability by combining Receptance Coupling SubstructureAnalysis(RCSA)andmachinelearningtechniques. The method utilizes RCSA to predict the tool tip FRF and a frequency-domainapproachtopredictthestabilityboundary. Updating is based on the binary test results (stable/unstable) and chatter frequency, if the test cut is unstable, as well as the user's risk preference. The machine learning model is implemented using Markov Chain Monte Carlo (MCMC) sampling. Postel et al. (2020) propose ensemble transfer learning in which the stability map is predicted using fully connected neural networks. The neural networks are pretrained with simulated data obtained from the analytical stabilitymodelandfine-tunedbyre-trainingwithexperimental test data to produce the final stability boundary. Yesilli et al. (2022) addressed the challenge of automating chatter detection in machining processes by exploring the potential of transfer learning. Their study evaluates the transfer learning capabilities of various chatter detection methods across turning and milling datasets. The findings revealed that Topological Data Analysis and Discrete Time Warping approaches can achieve comparable classification accuracies to time-frequency methods while offering the advantage of automation, particularly for scenarios involving limited data or small datasets of unique processes. Deng et al. (2023) introduces a novel approach for enhancing milling stability analysis. The paper proposes a multi-fidelity (MF) surrogate model combined with transfer learning to improve the accuracyofstabilitylobediagram(SLD)predictions.Themethod leverages both analytical stability modeling and experimental data to construct accurate stability models, resulting in morepreciselobeboundariesforselectingmachiningparameters while reducing the number of required experimental samples. Shanavas et al. (2023) introduces the application of supervised neural networks to learn stability maps from data, enhancing the prediction accuracy of machining stability maps beyond traditional models. Their study investigates the influence of hyperparameters on the learning process of neural networks, providing insights into



(a)



(b)



(c)

Fig. 2 **a** Stability map showing which cutting parameters will be stable. Cuts above the blue stability line will chatter and cuts below will be stable. **b** Stable cut surface. **c** Unstable cut surface

factors such as dataset size, learning rate, activation functions, and network architecture. Greis et al. (2023) introduces a physics-guided machine learning (PGML) for stability modeling in machining. This study examines how uncertain physics-based data can be employed to train a PGML stability model, subsequently updated with measured data, domain knowledge, and theory-based information. The paper presents four novel update strategies that enhance the PGML model's accuracy and reduce the need for extensive experimental measurements, showcasing its potential for approximating the true stability model for specific factory conditions.

Although the data-driven approaches provide improvements in specific circumstances, they have several limitations. For instance, machine learning approaches can lack physical interpretability, meaning that the stability maps do not expose the physical constraints that underlie the system's dynamics; in some cases, they even produce solutions that defy operational restrictions or the physical rules of stability limits (Greis et al., 2020; Postel et al., 2020). In addition, several approaches, exemplified by Friedrich et al. (2017), Cherukuri et al. (2019), and Karandikar et al. (2020), require a large number of sample data to learn the shape of stability maps, which can be expensive. While successful stability map prediction approaches have been proposed, a balance between the number of actual test experiments, physical interpretability, and prediction accuracy has not yet been achieved. To advance capabilities, this study introduces the Encoder GAN (EGAN) approach, which is based on Generative Adversarial Networks (GANs), for predicting stability maps using limited experimental data. EGAN is a modification of the traditional GAN that includes an additional encoder network to predict stability maps. The EGAN approachiscomposedoftheencoder,generator,anddiscriminator subnetworks. A similar GAN structure can be found in the work of Schlegl et al. (2019) but with a different application focus, namely anomaly detection. Furthermore, while Schlegl et al. trains their encoder separately from the generator and discriminator, our approach simultaneously trains all three components to improve overall performance. The input dataset for EGAN includes stability maps generated using a physics-based analytical stability model and the generator learnstomimicthephysics-basedmodel'sbehaviorbytaking a noise vector from the latent space and outputting a plausiblestabilitymap.Theencoderlearnstoprojectstabilitymaps back into the corresponding latent space parameter. The discriminator distinguishes generated/fake stability maps from sample maps in the input dataset. In the evaluation process, the trained encoder projects a stability map into the latent space to find the unique parameters that correspond to the map and the trained generator produces the corresponding stability maps based on the physics it learned. The approach is used to predict stability maps with a limited numberofcuttingtestexperimentsbyapplyingasimplenonphysics-based Bayesian updating approach to obtain partial knowledge about the target stability map. The hypothesis is that the perfectly trained EGAN has a regularized latent space that can map similar stability maps to closed points in the latent space, enabling the encoder to find the best parametersthatleveragealltheinformationitreceives,evenifthere is incomplete knowledge about the parameters and the stability maps. The novelty of the approach lies in the use of EGAN and the regularization property of the latent space to predict stability maps with limited information, which has the potential to reduce the number of cutting tests needed to predict stability maps accurately. We demonstrate the effectiveness of the proposed EGAN approach through extensive numerical simulations and experiments. We demonstrate the proficiency of the EGAN approach in effectively predicting stability maps, even when there is minimal data available regarding the unknown parameters. This is achieved through the consideration of wide normal and uniform distributions for uncertain parameters. A comprehensive comparison is also provided against state-of-the-art approaches, including ensemble transfer learning (Postel et al., 2020) and Bayesian learning (Karandikar et al., 2020).

The remainder of this paper is organized as follows: "Methodology"sectiondescribestheproposedEGANapproach used to predict the stability map. "Experimental verification" section evaluates the proposed approach. Finally, "Conclusion and future work" section concludes the study and discusses future work.

Model-based stability prediction

Chatter occurs due to the relative vibration between the cutting tool and workpiece during cutting (Schmitz & Smith, 2019). This vibration is imprinted on the workpiece surface (Fig.3). The next tooth encounters this wavy surface and experiences varying chip thickness h and, therefore, cutting force. During a stable cut, the cutting force and vibration will converge to an equilibrium that repeats for each tooth. In an unstable (chatter) cut, the force and vibration do not repeat from tooth to tooth and grow until the vibration is large enough that the tooth temporarily leaves the cut. This is referred to as the regenerative effect.

Theoccurrenceofchatterdependsontwomainfactors:the vibrationofthetool,governedbythetool'sFRF,andtheforce

involvedinthecuttingprocess,describedbythecuttingforce model. The tool tip FRF is described by a complex-valued, frequency-dependent matrix:

$$(\omega)] = \begin{bmatrix} FRF & [FRF_{xx}(\omega) & 0 \\ & & (1) \\ 0 & FRF_{yy}(\omega) & \end{bmatrix}$$

where x isthefeeddirectionand y istheorthogonaldirection in the plate of the cut. The cutting force is assumed to be proportional to the chip thickness $F = K_s bh$, where K_s is the specific cutting force coefficient and b is the chip width. This cutting force can be decomposed into the tangential force $F_t = \sin(\beta) K_s bh$ andnormalforce $F_n = \cos(\beta) K_s bh$, where β is the force angle.

The stability map can be estimated from these inputs using the zero-order approximation algorithm proposed by Altintas and Budak (1995). This method solves the periodiccoefficient delay differential equation by approximating the

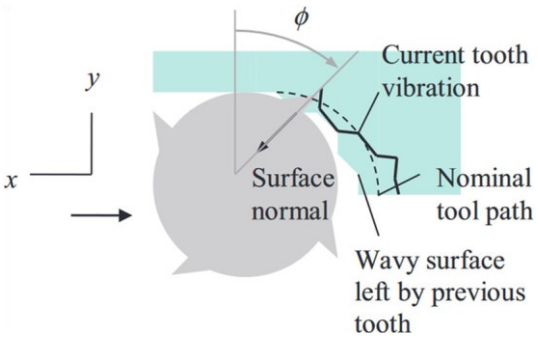


Fig. 3 Variable chip thickness due to tool vibration Schmitz and Smith (2019)

periodic cutting force as a matrix$[\alpha_{xx}\ \alpha_{xy}\alpha_{yx}\ \alpha_{yy}]$, where α_{ij} defines the amount that the average cutting force in the i direction will change for a unit displacement in the j direction. Please refer to Altintas and Budak (1995) for a full derivation.

Methodology

The purpose of the proposed study is to predict the stability map in milling operations using a novel deep learning approach named EGAN and limited experimental data. The primary distinction of the EGAN approach from previous approaches lies in the utilization of the GAN's structure for stability prediction. As far as our investigation shows, the proposed EGAN approach is the first study to employ a GAN-based architecture for predicting stability maps in machining processes. This incorporation of GAN architecture introduces a novel and innovative dimension to stability prediction that sets the proposed method apart from the existing approaches. By integrating an encoder network within the GAN framework, EGAN effectively bridges the gap between physics-based modeling and data-driven learning. This allows us to harness the advantages of GANs, such as generating plausible stability maps from latent space parameters, while also benefiting from the analytical insights of physics-based models. We believe that this pioneering utilization of GANs in the context of stability prediction not only contributes to the advancement of machining research but also opens avenues for future explorations at the intersection of deep learning and manufacturing.

In this section, the problem definition is first discussed. Then a brief background on the traditional GAN is provided. Next, the encoder GAN (EGAN), which is a leveraged version of the traditional GAN, is proposed for the purpose of the milling stability prediction. Then, common training challenges for GANs are discussed. Following that, a description of the state-of-the-art approaches that are used as benchmarks in this study are presented. Next, a grid search approach is introduced for the selection of cutting tests,
specificallyintendedforevaluationpurposes.Lastly,thegenera l framework of the proposed and benchmark approaches are presented.

Problem definition

Assuming a symmetric single degree-of-freedom (SDOF) system, there are five unknown parameters $\theta = (K_s, \beta, f_n, k, \zeta)$ to predict the stability map, where K_s is the specific cutting force coefficient, β is the force angle, f_n is natural frequency, k is the modal stiffness, and ζ is the modal viscous damping ratio (i.e., the SDOF FRF is approximated using these three parameters).

As described in "Introduction" section, an accurate estimation of the FRF and cutting force coefficients is required to estimate an accurate stability map. However, obtaining the FRF and cutting force coefficient information needs
specializedsensorsandcanbechallenginginindustry.Neverthe less, there exists a prior knowledge about these parameters. The prior represents the initial beliefs about parameters and incorporates all available information, whether it comes

from expert opinions, available experimental data, theoretical considerations, or analytical models. In this study, a prior distribution is first defined for these parameters $P(\theta)$ and N sampleparametersaregeneratedfromthisdistribution.Next, these sample parameters are used together with the physics model to generate prior/simulated stability maps, which are used as training data for the proposed approach. Finally, the trained networks apply limited experimental data to predict the target stability map.

Traditional GAN

The traditional GAN is a deep learning-based generative model that was initially proposed by Goodfellow et al. (2014). The primary goal of this approach is to generate synthetic but realistic images by learning the input data distribution by predicting features from an initially hidden representation, which is also called the latent space representation. The traditional GAN includes the generator and discriminator networks, competing with each other in an adversarial manner in the training process. Specifically, the traditional GAN is designed as a minimax game, where the generator and discriminator networks compete in a zero-sum gameframework,asoriginallyusedingametheory(Akcayet al.,2019).Thegenerator networklearnstogenerateanimage thatresemblesimagesintheinputdataset,whilethediscriminat ornetworklearnstodistinguishthegeneratedimagesfrom the input images.

Figure4 illustrates the scheme of the traditional GAN to generate handwritten digit numbers. In the traditional GAN, the generator (G) has a decoder network architecture and learns to capture the input data distribution (p_X) by generating plausible images from a latent space distribution (p_z), which is commonly a standard normal distribution. In contrast, the discriminator (D) adopts an encoder network architecture and works as a classification algorithm, reading an input image and labeling it as real or generated/fake. Both the generator and the discriminator networks are simultaneously trained through a two-player minimax game. During the training process, the generator aims to make the discriminator's loss larger. In other words, it tries to fool the discriminator by generating new plausible data (resembling the input data) so that the discriminator fails to detect them as fake images. Specifically, the original GAN solves the following minmax game (Goodfellow et al., 2014):

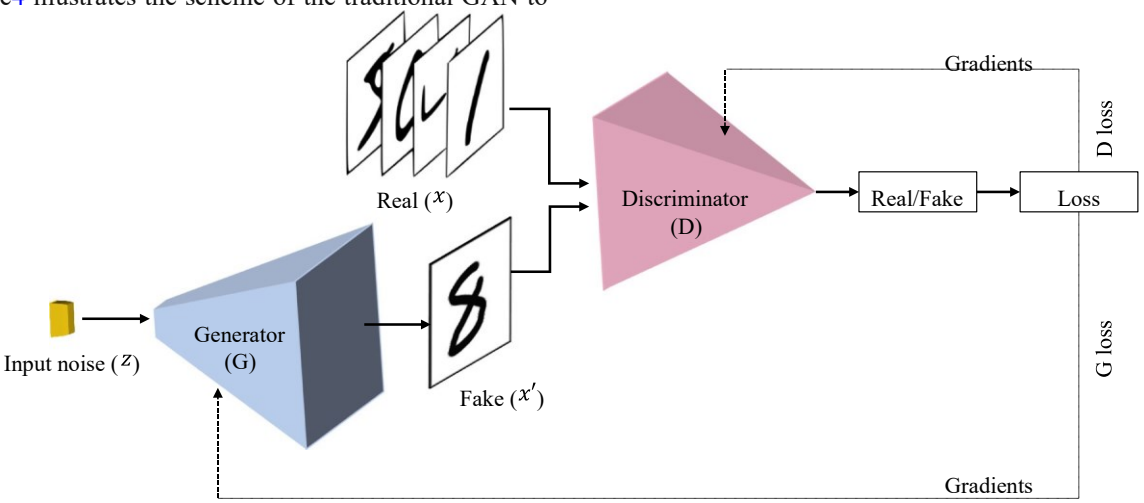


Fig. 4 Scheme of the traditional GAN

$$\min_G \max_D V(D,G) \tag{2}$$

where

$$V(D,G) = \mathbb{E}_{x\sim p_X}[\log(D(x))] + \mathbb{E}_{z\sim p_z}[\log(1 - D(G(z)))] \tag{3}$$

The discriminator is trained to maximize its outputs on input images (i.e., $\log(D(x))$; labeling input images as "real") and minimize its outputs on fake images (i.e., $\log(D(G(z)))$; labeling generated images as "fake"). In contrast, the generator is trained based on the discriminator's performance on fake images. The generator tries to minimize $V(G) = \log(1 - D(G(z)))$, or equivalently, maximize $V(G) = \log(D(G(z)))$, so it can fool the discriminator such that this network treats fake images as input images. Note that the generator is not able to control the discriminator's performance on input images (i.e., $\log(D(x))$).

Encoder GAN (EGAN)

Forpredictingstabilitymaps,theGANisadvancedbyadding encoder network (E) to the traditional GAN structure. The proposedapproachiscalledEncoderGAN(EGAN).Figure5 illustrates the scheme of the proposed approach.

The input/real dataset in EGAN includes stability maps generated using the physics-based analytical stability model, considering a distribution for the unknown parameters. The goal of the generator (G) in EGAN is to learn the physics underlyingthestabilitymapsintheinputdata.Thatis,ittakes avectorofnoisesfromthelatentspaceandoutputsaplausible stability map. The generator learns to mimic the physicsbasedanalyticalstabilitymodel'sbehavior.However,i nstead of taking the real parameters (θ), the generator takes some noise vector (z) from the latent space, which it is referred to as the latent space parameter in the remainder of the paper. It is important to clarify that the latent space is characterized as a Gaussian random noise vector. It serves as a transformed representation of parameters in an alternative space, devoid of direct physical interpretability. It is essential to underscore that the latent space lacks inherent interpretability.

On the other hand, the encoder E learns the opposite side of the generator. That is, it uses a stability map that is generated by the generator and returns the corresponding latent space parameter z. The discriminator (D) has the same role as it has in the traditional GAN. That is, it distinguishes the generated/fake stability maps output by the generator from sample maps in the input dataset. Through adversarial learning,thegeneratorlearnstogenerateralisticstabilitymaps to fool the discriminator and the decoder learns to find unique latent space parameters for each stability map.

The hypothesis is that if EGAN is perfectly trained, the generator connects each set of latent space parameters to a unique stability map, and, in contrast, the trained encoder projects back each stability map to a unique set of parameters. In addition, it is hypothesized that the perfectly trained EGAN has a regularized latent space, which means that the encoder canmapsimilarstabilitymapstoclosedpointsinthe latent space. These hypotheses are applied in the test process to predict the intended stability map with a limited number of cutting test experiments. For now, assume that complete knowledge of the target stability map is available, as seen in Fig.6.Thetrainedencodertakesthismapandprojectsitback into the latent space. The trained encoder finds the unique parametersthatcorrespondtothismap.Next,thetrainedgenerat ortakestheparametersthatthetrainedencoderfoundand produces the corresponding stability maps because it knows

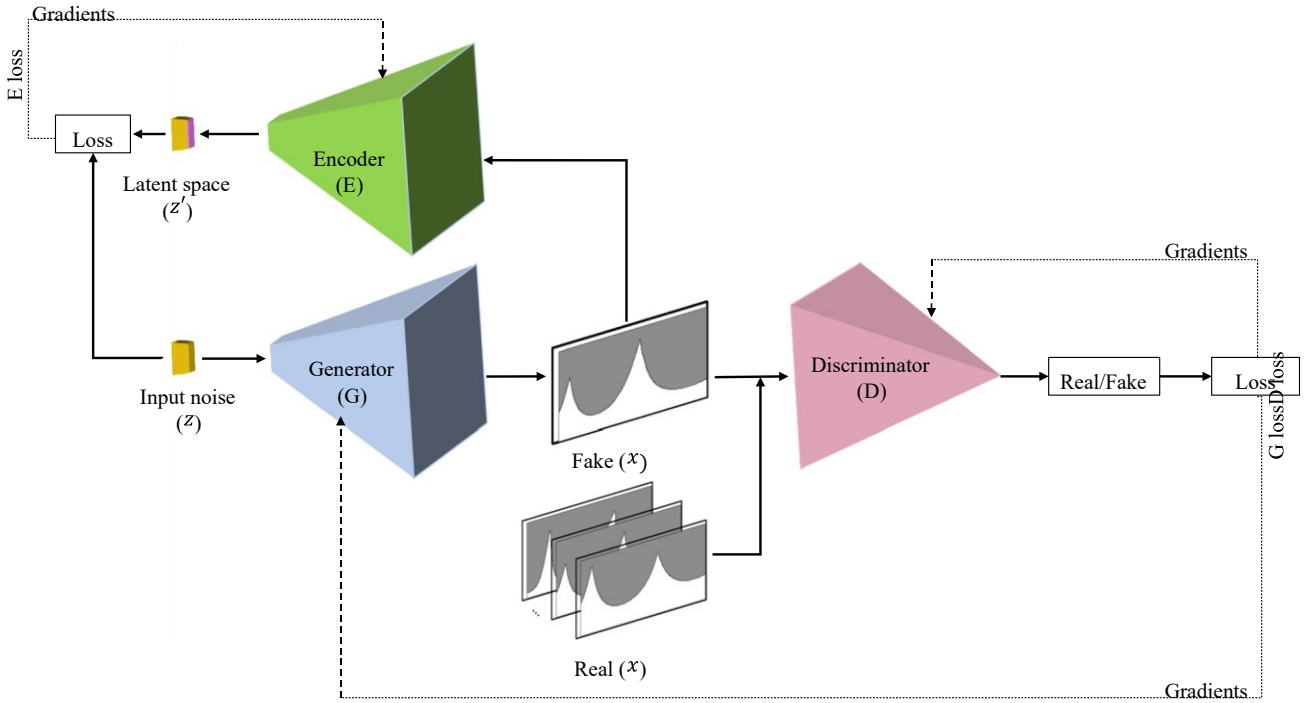


Fig. 5 Scheme of the EGAN

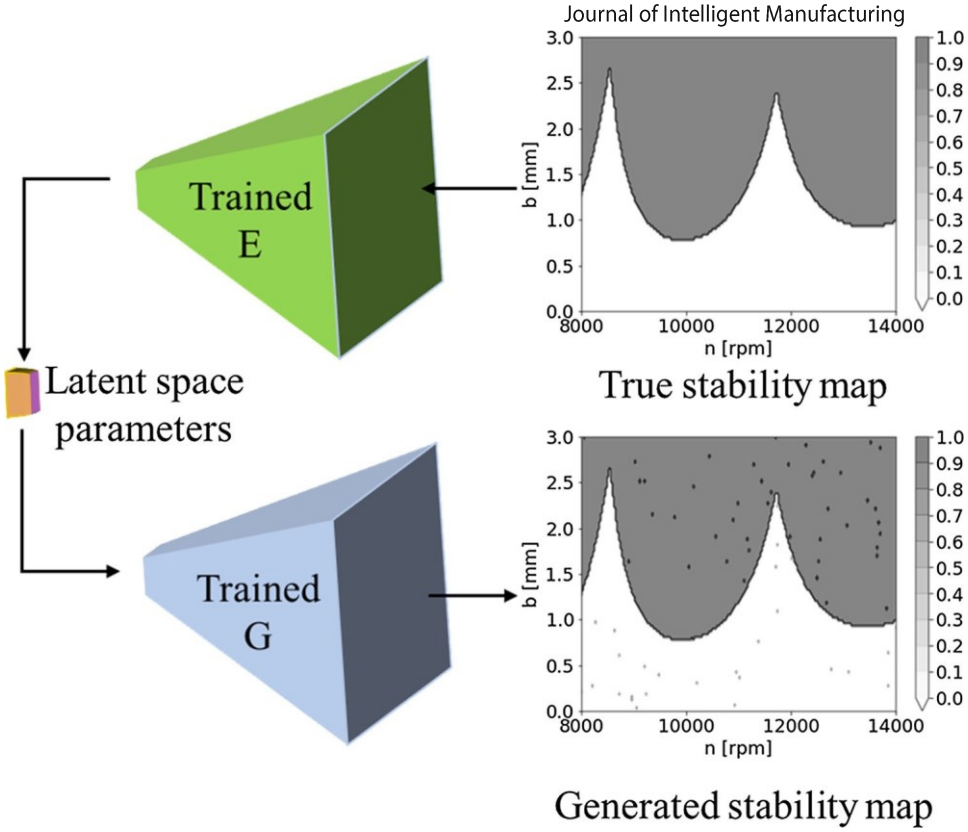


Fig. 6 Evaluation step in EGAN with complete knowledge of the stability map

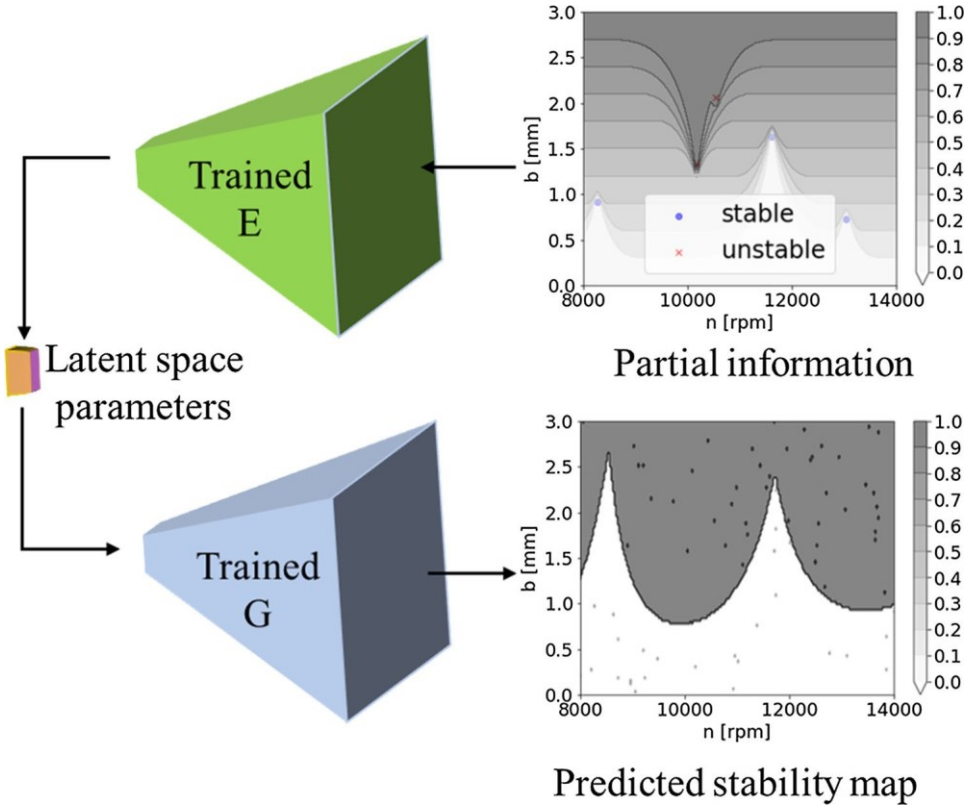


Fig. 7 Evaluation step in EGAN with partial information obtained from cutting tests

the physics. If both the generator and encoder are perfectly trained, the generator should output the exact map that the encoder receives. This is because, as hypothesized, there is only one vector of parameters for each stability map.

In practice, however, knowledge regarding the parameters (θ) and, consequently, the stability maps is incomplete. Hence, cutting tests are performed to obtain partial information about the target stability map and then this information is used as input to the proposed approach to predict the actual stability boundary. In this study, a simple non-physicsbased Bayesian updating approach is applied (as proposed in Karandikar et al. (2020)) to obtain partial knowledge (see Fig.7) about the target stability map using a limited number of experiments. This approach is discussed in the benchmark section("Bayesianlearning"section).Itishypothesizedthat, given this partial knowledge of the stability map, the perfectly trained encoder will find the best parameters in latent space that incorporates all the information that it receives. If there is sufficient information, it can find the exact vector of parameters in the latent space that corresponds to the target stability map and, as a result, the generator can then return the target stability limit. However, if the information is not sufficient, the encoder may not be able to find the true latent space parameters, but at least it can find parameters that are close enough to the true parameters. Consequently, the generator may not be able to reconstruct the target stability map, but it can output a map that is close enough to the target stability map. This behavior is explained using the regularization property of the latent space.

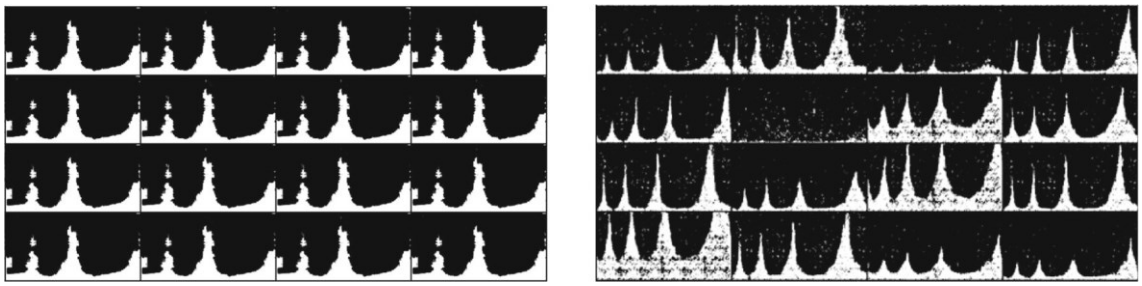
To train the model in a way that enables the hypotheses to be tested, the loss functions of each sub-network are formulated as follows:

Generator Loss: The generator network G is trained to learn the physics underlying the input dataset and generate unique stability maps for each set of latent space parameters. To do so, the generator needs to fool the discriminator to be able to generate plausible stability maps. Hence, the generator's loss should compute how effectively the discriminator distinguishes generated/fake stability maps from input stability maps. To capture this, the generator's loss is calculated as:

$$L_G = \mathop{\mathbb{E}}_{z\sim p_z}[\log(D(G(z)))] \tag{4}$$

Encoder Loss: The encoder network E should learn the opposing side of the generator. That is, it should be able to project the stability maps generated by network G back into the latent space to find the corresponding latent space parameters. The encoder can learn this information by calculating dissimilarities between the input parameters of the generator (z) and the corresponding reconstructed

parameters $z = E(G(z))$. Therefore, the encoder's loss is calculated as:



(a) Mode collapse (b) Instability of training

Fig. 8 Examples of mode collapse and training instability in the EGAN approach

$$L_E = \mathbb{E}_{z \sim p_z} [\|z - E(G(z))\|_1 \quad (5)$$

Discriminator Loss: In adversarial learning, the discriminator is trained to classify both the real data and the fake data from the generator. It penalizes itself for incorrectly categorizing a fake instance (generated by the generator) as real or a real instance as fake by maximizing the function shown in Eq.(6).

$$L_D = \mathbb{E}_{x \sim p_X} [\log(D(x))] + \mathbb{E}_{z \sim p_z} [\log(1 - D(G(z)))] \quad (6)$$

Common training problems in GANs

The traditional GAN in its original form suffers from two major limitations including mode collapse and nonconvergence (or training instability). Mode collapse is known as the lack of image diversity in GAN training and happens when the generator learns to associate multiple input z values with a single output point. Non-convergence mainly happens when the equilibrium between the discriminator and the generator is not found. For instance, when the discriminator perfectly distinguishes between the fake and real images and starts to reject the generated samples by the generator, there is no longer enough information for the generator to learn from, which causes the vanishing gradient problem for the generator.

In recent years, empirical approaches have been introduced to enhance the training efficiency of GANs regarding mode collapse and/or training instability. For instance, the Wasserstein GAN (WGAN) proposed in Arjovsky et al. (2017) has demonstrated its capacity to mitigate both of these issues. In addition, past studies (Ham et al., 2020; Munjal et al., 2020; Rosca et al., 2017; Bang & Shim, 2021; Chong et al., 2020; Bang & Shim, 2018; Lazarou, 2020) have shown that autoencoder-based GANs generally have the ability to prevent mode collapse. One of the most effective approaches to overcome non-convergence (i.e., instability in training) is Deep Convolutional GAN (DCGAN) proposed by Radford et al. (2015), where convolutional layers replace fully connected layers. In addition, Chakraborty et al. (2018) mention that using batch normalization in GAN's structure can reduce the problem of training instability, which is caused by poor initialization. Salimans et al. (2016) show that the feature matching loss can address the instability of training caused by over-training on the discriminator response. Goodfellow et al. (2014) suggest training the discriminator for $n_d > 1$ steps every time the generator is trained for one step. This would help balancing the power/optimization of the discriminator and generator networks. Goodfellow (2016) also recommend to balance the generator and discriminator by optimizing the model size.

Figure 8 provides examples of the mode collapse as well as the training instability problems in EGAN. Specifically, this figure presents a complete batch of 16 stability maps, which are generated using the generator network by inputting 16 random Gaussian noise samples. As seen in Fig. 8a, it is apparent that the generator network suffers from the mode collapse problem and fails to produce diverse and realistic stability maps. In addition, Fig. 8b demonstrates that the generator network is affected by training instability, which leads to the generation of noisy stability maps.

Although many approaches have been proposed to overcome mode collapse and instability of training problems, they have not been completely solved. In this study, existing approaches are applied to address both problems, which are explained later in the results section.

Benchmarks

In this section, two state-of-the-art machine learning approaches are described, including Bayesian learning and ensemble transfer learning approaches, that have been successfully applied for stability prediction. These two approaches are used as benchmarks in this study.

Bayesian learning

The Bayesian learning approach, which is proposed by Karandikar et al. (2020), finds the stability boundary in milling without knowledge of the underlying tool dynamics or cutting force coefficients. In this approach, a prior for the stability map is first identified. The prior is the current beliefs about the parameters that are formed based on all available data. Second, the prior probability is updated using the Bayes' rule and experimental data.

In this study, the prior probabilities are established using the basic knowledge that high axial depths are more likely to be unstable at a selected spindle speed. To define the prior, the space of the stability map is first divided into several grid points. Then, the prior probability of stability ($p(s_G)$) for each grid point G on the map is defined using the prior knowledge (see Fig.9a). As a result, each grid point has a $p(s_G)$ probability of stability and a $p(u_G) = 1 - p(s_G)$ probability of instability (Fig. 10).

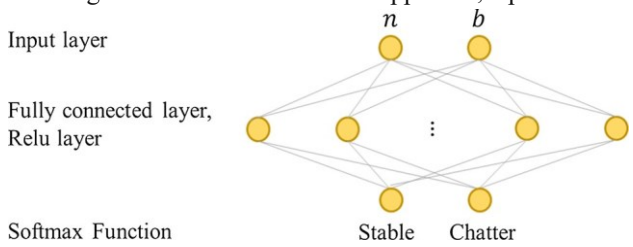
The goal is to update the probability of all grid points given a stable (+) or unstable (−) result at the test point T. Equation(7) shows the Bayes' rule to update the probability at grid point G based on a stable result at test point T.

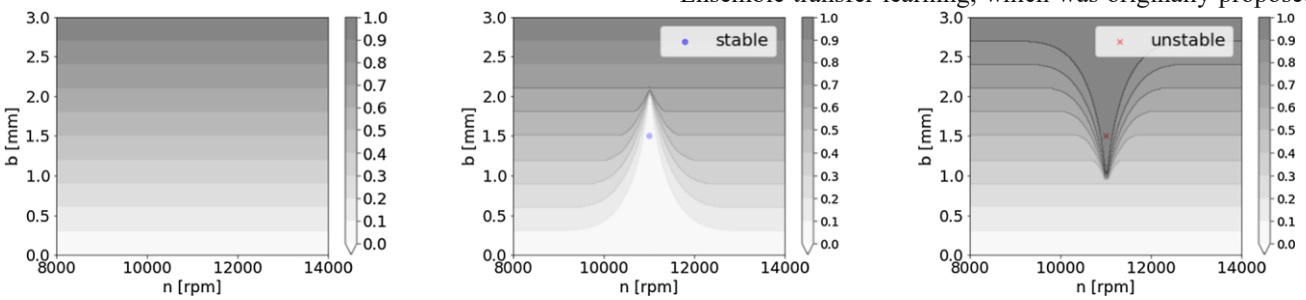
$$p(s_G|+_T) = \frac{p(+_T|s_G)p(s_G)}{p(+_T)}, \tag{7}$$

$$p(+_T) = p(+_T|s_G)p(s_G) + p(+_T|u_G)p(u_G) \tag{8}$$

Based on expert knowledge of the stability behavior, the likelihood probabilities $p(+_T|s_G)$ and $p(+_T|u_G)$ are defined. For more details please refer to Karandikar et al. (2020).

Figure 9b and c present the posterior probabilities updated using these equations for a stable and unstable test result, respectively.





(a) Prior probability of stability map

(b) Posterior probability of stability given a stable result at 11,000 rpm, 1.5 mm

(c) Posterior probability of stability given an unstable result at 11,000 rpm, 1.5 mm

Fig. 9 Stability prior and updatet posterior given stable and unstable results

Fig. 10 Neural network architecture in ensemble transfer learning

Ensemble transfer learning

Ensemble transfer learning, which was originally proposed by Postel et al. (2020), is a deep learning-based approach that is designed to predict milling stability maps in milling by utilizing simulated and experimental data. In this approach, simulated data is utilized as the starting point for training feed-forward neural networks to learn the concept of stability maps. Next, experimental data are used to fine-tune the pre-trained neural networks so that these networks can adapt to the actual behavior of the system. The ensemble transfer learning framework includes six steps, as follows:

(1) *Parameter sampling* In this step, the uncertain parameters are sampled N_{net} times from their distributions and then fed to an existing stability model to generate the corresponding stability maps. This step results in N_{net} simulated stability maps.

(2) *Generating training datasets* Each stability map generated in Step 1 is divided into several grid points to construct N_{sim} artificial cutting tests (n_i,b_i). As a result, N_{net} training datasets are constructed in this step, each of which contains N_{sim} cutting tests.

(3) *Stability evaluation* In this step, the stable or unstable labels c_i for all pairs of spindle speed and cutting depth (n_i,b_i) in each training dataset are determined by

utilizing the corresponding stability map generated in Step 1.

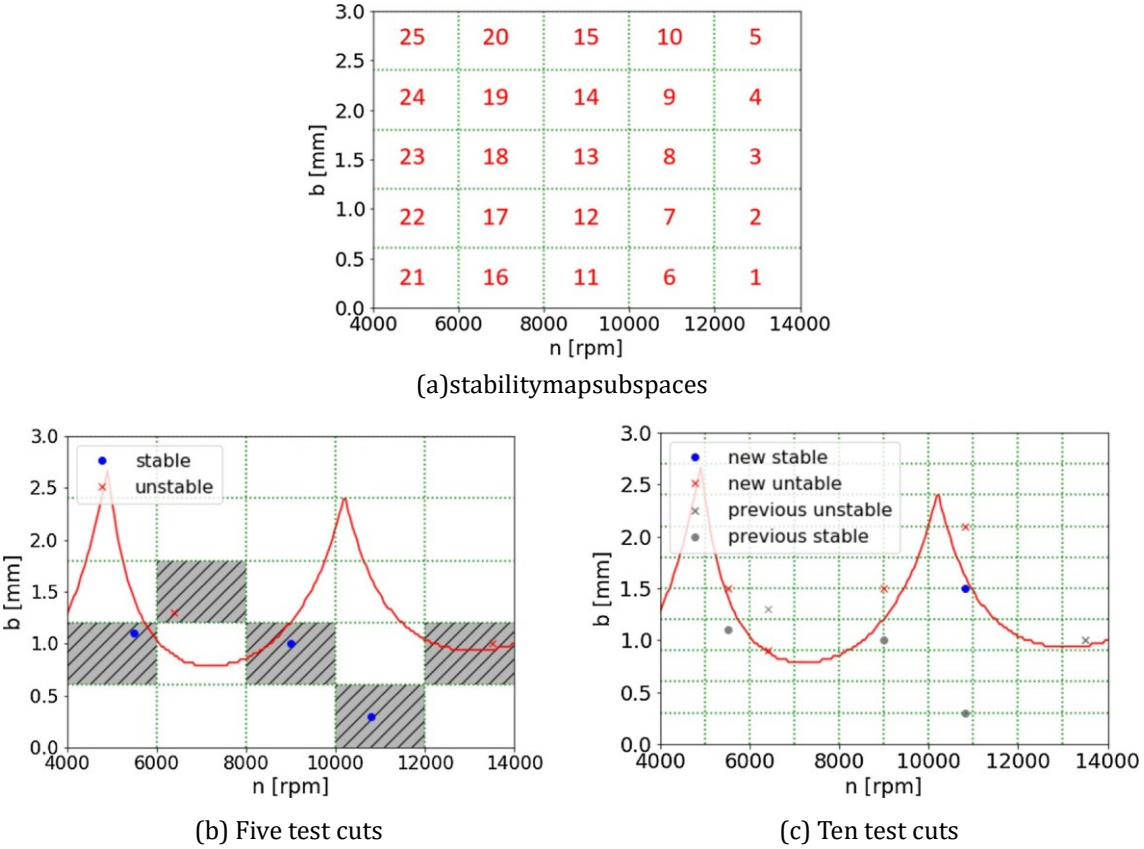
(4) *Pre-training* Each generated dataset from Step 2 and corresponding output labels fromStep3areusedtopre-train a fully connected neural network. As a result, N_{net} neural networks are trained to learn the shape of the each of the stability map in the training dataset. The structure of these neural networks is presented in Fig.31.

(5) *Fine tuning* In this step, each pre-trained neural network in Step 4 is fine-tuned by utilizing the experimental cutting tests. As a result, the pre-trained neural network that had learned the general shape of the stability boundary is now adapted to the actual stability behavior using the experimental data.

(6) *Prediction* Finally, each of the fine-tuned networks in Step 5 is used to predict the target stability map. Next, a truncated mean approach is used to average the stability maps obtained from N_{net} fine-tuned networks to get the final stability map.

Cutting tests selection

This section outlines the approach for selecting five and ten cutting tests, which are conducted for evaluation. This study initiates by randomly selecting five cutting tests from the stability map space (i.e., spindle speed and axial depth of cut) to predict the actual stability map. Subsequently, an additional five cutting tests are selected based on the previous set of five chosen points to form a total of ten test points, to increase the accuracy of the stability map prediction.



(a)stabilitymapsubspaces

(b) Five test cuts

(c) Ten test cuts

Fig. 11 Grid search to select five and 10 test cuts in the numerical analysis

The initial selection of five test points is accomplished through the use of a grid-based approach, which is detailed below. First, the stability map space is partitioned into 25 subspaces by equally dividing the range of spindle speed and axial depth of cut into five parts (see Fig.11a). Next, the first test point is randomly selected from the second subspace. Note that the second subspace is always used as the starting point in all setups. Then, the next subspace is selected from the second range of spindle speed (i.e. from subspaces 6, 7, 8, 9, and 10) considering the stability behavior of the current test result. If the current test result is stable, the subsequent subspace is selected from the upper range of axial depth of cut. Conversely, if the current test result is unstable, the next subspace is selected from the lower range of axial depth

ofcut.Once thesubspace isselected, thenext testpoint is chosen randomly from the selected subspace. This process is repeated until all five test points are selected. In this way, this approach ensures that at least one test from each range of spindle speed is selected. Figure11b presents a sample path for the selection of five cutting tests from a known stability limit.

To obtain 10 test cuts, the initial selection of five is extended by including an additional set of five. The selection begins by partitioning the axial depth of cut into 10 equal parts, thereby subdividing the stability map space into more subspaces. Next, a test cut is selected from the previous set of tests and, depending on the stability of this selected test cut, tests with larger axial depth of cuts but the same spindle speed are chosen if stable, or tests with lower axial depth of cuts but the same spindle speed are chosen if unstable. Figure11c illustrates a sample path to extract 10 test points from a known stability limit. As depicted in the figure, this approachyieldstheselectionofpointsthatareinproximityto the stability boundary, thereby providing adequate information for both the EGAN and other approaches to accurately predict the target stability map.

General framework of the proposed and benchmarks approaches

The general framework of the proposed and benchmarks approaches is presented in Fig.12. In the first step, N sample parameters $(\{\theta^i = (K_s^i, \beta^i, f_n^i, k^i, \zeta^i) | i = 1, \ldots, N\})$ are generated from their distribution $P(\theta)$. These parameters are input to the physics model to find the stability limit b_{lim} for each axial depth of cut n and, as a result, the corresponding stability map. This step results in N simulated stability maps. These simulated stability maps are then used as training samples to train the EGAN and ensemble transfer learning approaches in the second step. Note that in the EGAN approach, the last M = 100 trained networks are selected to make prediction. Next, in the third step, a small number of test cuts are selected based on the grid search approach proposed in "Cutting tests selection" section. As seen in the figure, the cutting test points are augmented by establishing that for a stable test, all test points at the same spindle speed and a lower axial depth of cut will also exhibit stability. Conversely, if a test is unstable, then all test points with the same spindle speed and a higher axial depth of cut will also exhibit instability.

Finally, in the fourth step, the target stability map is predicted using the Bayesian learning, ensemble transfer learning, and EGAN approaches. The cutting tests are used by the Bayesian learning approach to predict the target stability map. The augmented cutting tests are used to fine tune the pre-trained neural networks. Next these fine-tuned neural networks are used to predict the stability map. Lastly, the partial information obtained by the Bayesian learning approach is used as input for the M trained encoders and generators in EGAN to predict the target stability map. Each trained encoder receives the partial information as input and attempts to find the best latent space parameters that cover the maximum amount of information it receives. Next, each of the trained generators is used to predict the actual stability map from the latent parameters that the encoder found. Lastly, the prediction is obtained by taking the average of the M outputs of the generators.

Experimental verification

Numerical experiments

In this section, numerical experiments are conducted to evaluate the proposed and benchmark approaches to identifying a known stability map. A description of the setups for the numerical experiments is first provided. Next, the training performances of the EGAN and ensemble transfer learning approaches are presented. Finally, a comparison between the EGAN, ensemble transfer learning, and Bayesian learning approaches using numerical experiments is provided.

Experiments setup

ToinvestigatetheefficiencyoftheproposedEGANapproach for milling stability prediction under different target conditions, three tool setups with different process and modal parameters are selected, and their specific information is listed in Tables 1 and 2. Specifically, Table 1 presents process parameters for these three tool setups, including the tool diameter, number of teeth, feed, feed direction, and radial immersion, which refers to the depth of cut as a percentage of the tool diameter. As seen in this table, a climb-milling machining process is considered with tool diameters in the range of 12.7 mm to 25.4 mm, three or four teeth, and radial, and radial immersion in the range of 25% to 75% for different setups. These simulations are not based on specific work materials, but the cutting force values are generally in the range common for aluminum and other soft metals.

Table 2 presents the uncertain process and modal parameters with their respective distributions for the three tool setups. As seen in Table 2, in Setups 1 and 2, a SDOF system is considered with five unknown parameters related to the cutting coefficients and tool tip dynamics, including K_s, β, k, f_n, and ζ. The entries for Setups 1 and 2 can be

considered as a series of five univariate normal distributions with mean values of $\mu_1,\mu_2,\mu_3,\mu_4,\mu_5$ and standard deviations of $\sigma_1,\sigma_2,\sigma_3,\sigma_4,\sigma_5$. Generating sample parameters from each of these distributions gives a series of values $\theta^i = (K_s^i, \beta^i, f_n^i, k^i, \zeta^i)$. However, it is more beneficial to consider the inputs as a single five-dimensional multivariate normal distribution $N5(\mu_s, \Sigma_s)$with a mean vector μ_s and covariance matrix $_s$:

$$\mu_s = [\mu_1, \mu_2, \mu_3, \mu_4, \mu_5] \tag{9}$$

$$\Sigma_s = \begin{bmatrix} \sigma_1^2 & 0 & 0 & 0 & 0 \\ 0 & \sigma_2^2 & 0 & 0 & 0 \\ 0 & 0 & \sigma_3^2 & 0 & 0 \\ 0 & 0 & 0 & \sigma_4^2 & 0 \\ 0 & 0 & 0 & 0 & \sigma_5^2 \end{bmatrix} \tag{10}$$

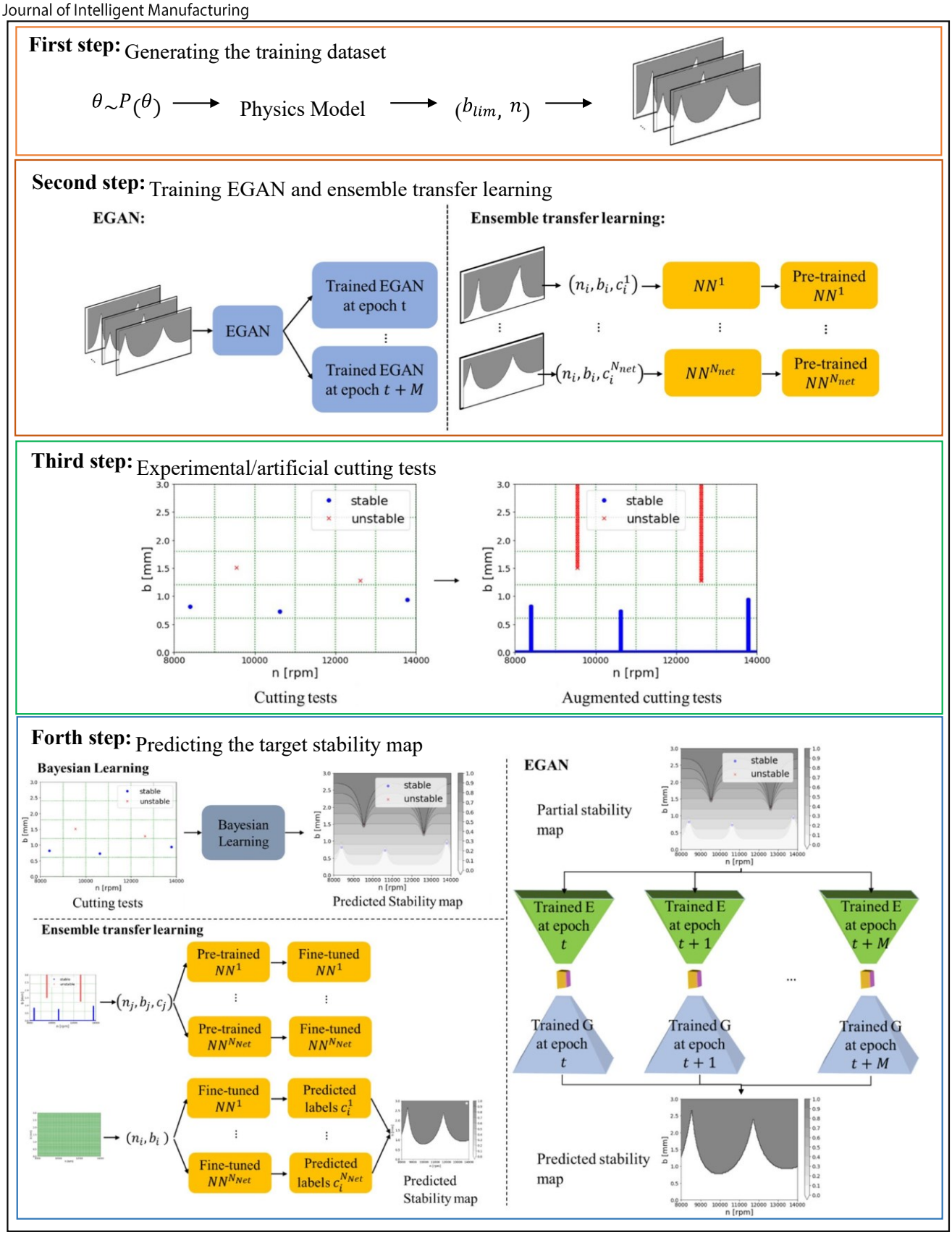


Fig. 12 General framework of the proposed and benchmarks approaches

Table 1 Process parameters for three tool setups

	Setup 1	Setup 2	Setup 3	Unit
Tool diameter	25.4	25.4	12.7	mm
Number of teeth	4	4	3	–
Feed	0.1	0.1	0.1	mm/tooth
Feed direction	Climb	Climb	Climb	–
Radial immersion	25%	50%	75%	–

InSetup3,a2DOFwitheightunknownparametersrelated to the cutting coefficients and tool tip dynamics is considered.

$$\mu_{s3} = [4 \times 10^8, 68, 1301, 6.64 \times 10^6, 0.049, 1659, 6.75 \times 10^6, 0.03] \tag{11}$$

$$\Sigma s3 = \begin{bmatrix} 1 \times 10^{16} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 46.24 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 5.3 \times 10^3 & -1.2 \times 10^8 & 0.67 & 1.3 \times 10^4 & 1 \times 10^8 & -1.5 \\ 0 & 0 & -1.2 \times 10^8 & 5.7 \times 10^{12} & -2.1 \times 10^4 & -3.6 \times 10^8 & -3.8 \times 10^{12} & 6.1 \times 10^4 \\ 0 & 0 & 0.67 & -2.1 \times 10^4 & 1.1 \times 10^{-4} & 1.72 & 1.7 \times 10^4 & -1.9 \times 10^{-4} \\ 0 & 0 & 1.3 \times 10^4 & -3.6 \times 10^8 & 1.72 & 3.5 \times 10^4 & 2.8 \times 10^8 & -4.57 \\ 0 & 0 & 1 \times 10^8 & -3.8 \times 10^{12} & 1.7 \times 10^4 & 2.8 \times 10^8 & 2.8 \times 10^{12} & -4.1 \times 10^4 \\ 0 & 0 & -1.5 & 6.1 \times 10^4 & -1.9 \times 10^{-4} & -4.57 & -4.1 \times 10^4 & 8.3 \times 10^{-4} \end{bmatrix} \tag{12}$$

The uncertainty of these parameters is captured through an eight-dimensional multivariate normal distribution $N_8(\mu_{s3}, \Sigma_{s3})$ with mean vector μ_{s3} and covariance matrix Σ_{s3}; the subscript 3 denotes the setup number. The values for the vector μ_{s3} and matrix Σ_{s3} are presented in Eqs.(11) and (12).

out chatter b_{lim} for some spindle speeds n. Determining this value for each spindle speed and for each parameter results in the probabilistic stability map depicted in Fig.13. The

Each tool setup has two target stability maps associated with it. These target stability maps represent specific desired outcomesforthecuttingprocessunderinvestigation.Toelaborate further,thesetargetstabilitymapsareextractedthrough aprocesswhereparametersaregeneratedfromtheparameter distributions outlined in Table 2. For each generated parameterset,thephysicsanalyticalmodelisemployedtoextract the corresponding stability map. These extracted stability maps, termed"targetstabilitymaps,"serveasbenchmarkreferences for evaluating the predictive performance of stability prediction methods, such as the EGAN approach. Table 2 presents the parameter values and corresponding stability maps for the two targets considered in each setup. In this table, the target stability maps are shown along with the stability map that is obtained using the mean of the normal distribution, which is labeled "Prior Mean" in the map.

grayscale level of each point (n,b) is the probability of stability, which is calculated using the following equation:

$$P_{stability} = \frac{1}{N} \sum_{i=1}^{N} b_{lim}(n, \theta_i) < b \tag{13}$$

The probability of stability at a specific point indicates the number of stability maps that forecast stable cutting test at that point. Figure13 shows the resulting probabilistic stabilitymap along withtheprioriand target stabilitymaps foreach setup. As expected, the probabilistic stability map could represent the stability map that is related to the prior mean with a good approximation; however, it does not represent the stability maps that are related to Targets 1 and 2. For instance, Target 2 in Setup 2 completely contradicts the probabilistic

Toshowhowcloselythepriorwouldrepresenteachtarget, theprobabilisticmapisconstructed.Todoso,thepriordistribution listed in Table 2 is used to generate N = 4000 sample parameters, which are used as input to the physics model to predict the stability maps. For each sample parameter θ, the physics model computes the maximum cutting depth withstability map. Note that the simulated stability maps that are applied here to draw the probabilistic stability map are also used as the training dataset to train the EGAN and transfer learning approaches.

The rationale behind generating the probabilistic map and subsequently comparing it with the target stability maps in Fig.13 is to underscore a significant observation: the training dataset, instrumental in the construction of the probabilistic stability map, does not inherently encompass the intricate characteristics of the target stability maps employed during the evaluation process. This distinction

highlights the challengefacedbytheapproachinaccuratelypredictingthetarget stability maps, given the dissimilarity between the training and evaluation datasets.

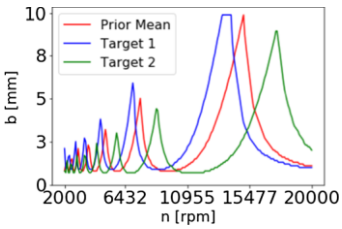
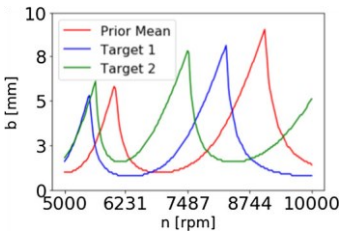
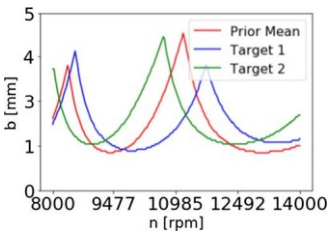
Training performance

This section presents the training performance of the EGAN and transfer learning approaches.[1] It should be noted that the Bayesian learning approach does not include a distinct training step. To train these approaches, 4000 sample stability maps are generated using the uncertainty distribution provided in Table 2. These samples are then employed to train the EGAN approach and pre-train the neural networks intheensembletransferlearningapproach.Thisenablesboth approaches to acquire a comprehensive understanding of stability maps.

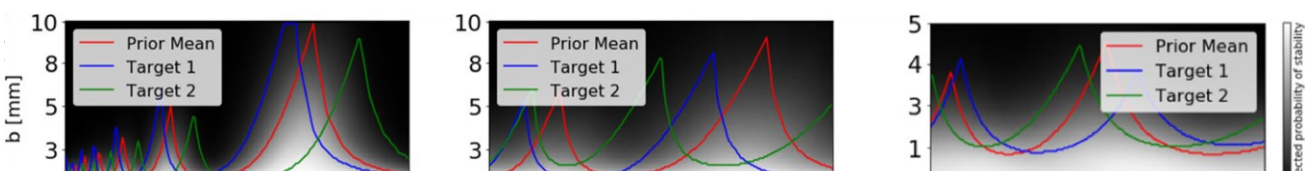
EGAN Network Architecture Per the discussion in "Common training problems in GANs" section, deep con-

[1] The code is available at https://github.com/srezaei90/GANs-topredict-stability-maps-in-milling-machining.git.

Table 2 Uncertain process and modal parameters with their respective distributions for three tool setups

	Parameter	Distribution (N(μ,σ))	Target 1	Target 2	Stability map
Setup 1	$K_s(N/m^2)$	$N(8 \times 10^8, 1.6 \times 10^8)$	5.2×10^8	10.2×10^8	
	β (degree)	N(68,6.8)	63	74	
	f_n (Hz)	N(1000,100)	929	1164	
	k (N/m)	$N(8 \times 10^6, 1.6 \times 10^6)$	6.16×10^6	9.94×10^6	
	ζ	N(0.03,0.006)	0.027	0.027	
Setup 2	$K_s(N/m^2)$	$N(6 \times 10^8, 1.2 \times 10^8)$	6.67×10^8	4.99×10^8	
	β (degree)	N(68,6.8)	67	58	
	f_n (Hz)	N(1200,120)	1096	1484	
	k (N/m)	$N(5 \times 10^6, 1 \times 10^6)$	5.1×10^6	4.28×10^6	
	ζ	N(0.02,0.004)	0.017	0.024	
Setup 3	$K_s(N/m^2)$		6×10^8	3.7×10^8	
	β (degree)		68	57	
	f_n1 (Hz)		1286	1272	
	k1 (N/m)	$\mathcal{N}(\mu_{s3}, \Sigma_{s3})$ $\mathcal{N}(\mu_{s3}, \Sigma_{s3})$ $\mathcal{N}(\mu_{s3}, \Sigma_{s3})$ $\mathcal{N}(\mu_{s3}, \Sigma_{s3})$ $\mathcal{N}(\mu_{s3}, \Sigma_{s3})$ $\mathcal{N}(\mu_{s3}, \Sigma_{s3})$ $\mathcal{N}(\mu_{s3}, \Sigma_{s3})$ $\mathcal{N}(\mu_{s3}, \Sigma_{s3})$	4.72×10^6	8.2×10^6	
	ζ1		0.041	0.040	
	f_n2 (Hz)		1694	1584	
	k2 (N/m)		7.90×10^6	5.50×10^6	
	ζ2		0.033	0.038	

Two target stability maps in each setup are considered for prediction in the numerical experiments



volutional layers with batch normalization are used for the sub-networks in the architecture of the EGAN approach to overcome mode collapse and training instability. The models (i.e., the generator, encoder, and discriminator) are initialized with the hyperparameter values suggested in (Radford et al., 2015), but the parameters that are most important to learning, including number of hidden layers, activation functions, learning rate, batch size, and latent space dimension are updated to get a perfectly trained generator and encoder. To balance the power of the discriminator and generator

networks, the discriminator's size is decreased to two convolutional layers and the generator's size is increased to eight convolutional transpose layers. In addition, it is observed that increasing the number of steps for training the discriminator (n_d) per iteration of generator training can aid in reducing noise in the generated images. However, the optimal value of n_d may vary depending on the specific tool setup being used. Therefore, it is recommended to experiment with different values of n_d, ranging from 1, and iteratively adjust it to find the best-performing model for a given tool setup. Furthermore, one can choose between the sigmoid and hyperbolic tangent (tanh) activation functions in the last layer of the generator to reduce the noise in the generated images. Table 3 provides detailed information on the layers of the generator, encoder, and discriminator networks, including the corresponding activation functions, optimizer used for each network, and relevant hyperparameters. Note that the optimal dimensionality of the latent space is set to four.

EGAN Training Performance The performance of the achieved generator and encoder networks during the training process on Setup 2 is illustrated in Figs. 14 and 15. Specifically, Fig. 14 depicts the progression of the generator's ability to produce plausible stability maps throughout the training process. As depicted in the figure, the generator initially produces random noise in the first few epochs. However, as training progresses, it acquires the ability to fool the discriminator and ultimately generate plausible stability maps. Despite the generator's ability to produce a diverse range of stability maps, it still exhibits some level of noise in the generated output. This suggests that the issue of training instability has not been entirely resolved, even though the generator does not suffer from mode collapse.

Figure 15 depicts the advancement of both the encoder and generator's capability to reproduce a given stability map throughout the training process. More specifically, this figure demonstrates how the encoder can find the unique latent space parameters that correspond to the input stability map as it undergoes the training process. As explained in the Methodology section, a perfectly trained encoder and generator should result in a reconstructed stability map ($G(E(x))$) that is either identical or very similar to the input stability map x. The input stability map depicted in this figure is derived from the "Prior Mean" utilized in Setup 2. As illustrated in the figure, during the initial epochs, the encoder is unable to identify the latent parameter that corresponds to the input map. This outcome is to be expected, given that the generator possesses only a limited understanding of stability maps in the initial epochs. Consequently, the encoder, which relies on the generator's outputs for learning purposes, is unable to accurately connect the input stability map to the latent space. As the training progresses, the generator gains a deeper understanding of the physics underlying stability maps, and as a result, the encoder becomes more adept at accurately connecting the input stability map to the latent space. As depicted in the figure, at epoch 100, the encoder and generator are able to successfully reconstruct the input stability map with a high degree of accuracy.

Ensemble Transfer Learning Architecture In ensemble transfer learning, it is necessary to pre-train N_{net} fully connected neural networks on N_{net} simulated stability maps so that each neural network can learn the shape of each stability map. However, the entire training dataset, including 4000 simulated maps, cannot be used to train 4000 neural networks, as it would be computationally expensive. Hence, a subset is selected from the training dataset containing 200 sample stability maps to train $N_{net} = 200$ neural networks. Figure 16 illustrates the probabilistic stability map for each setup that is obtained by 200 simulated stability maps that are used to pre-train neural networks. This figure shows that even though only 200 sample stability maps are used, the resulting probabilistic map is very close to the one that is made with 4000 sample stability maps (see Fig. 13).

To achieve effective transfer learning, it's crucial to use a neural network architecture that is appropriate for replicating stability behavior. To accomplish this, hyperparameter tuning is done by evaluating the network's performance on a simulated dataset. The number of hidden layers and the number of nodes in each layer are set as hyperparameters to determine an acceptable network structure. Table 4 provides detailed information on the layers of all N_{net} neural networks, and their corresponding learning rate, optimizer, and training epochs. Note that all N_{net} neural networks share a common network architecture.

Ensemble Transfer Learning Training Performance Fig. 17 shows the output prediction of three pre-trained neural networks, which are trained on three sample stability maps (shown as red lines) from the training dataset of Setup 1. As this figure shows, all three neural networks are perfectly pretrained to replicate the behavior of the respective stability maps.

EGAN versus benchmarks

This section utilizes the proposed EGAN approach and benchmarks including the ensemble transfer learning and Bayesian learning to make predictions by utilizing numerical tests derived from a known stability map. The objective is to evaluate the accuracy of the stability

predictions made by the proposed and benchmark approaches through a comparison of their forecasts against a known stability map. The primary focus is to demonstrate the proficiency of the EGAN approach in predicting the stability map, even with a limited number of numerical tests (limited to five and 10 numerical test cuts). Through these methods, a thorough assessment of the stability predictions made by different approaches and their potential for practical applications is provided.

Table 3 Architecture of the EGAN approach

Layer	Kernel size	Stride size	Panel size	Output size
Generator				
Input	–	–	–	(4, 1, 1)
ConvTran (Batchnorm/ReLU)	(6,3)	(1,1)	(0,0)	(4096, 6, 3)
ConvTran (Batchnorm/ReLU)	(2,2)	(1,1)	(0,0)	(2048, 7, 4)
ConvTran (Batchnorm/ReLU)	(2,2)	(1,1)	(0,0)	(1024, 8, 5)
ConvTran (Batchnorm/ReLU)	(2,2)	(2,2)	(1,1)	(512, 14, 8)
ConvTran (Batchnorm/ReLU)	(2,2)	(2,2)	(1,1)	(256, 26, 14)
ConvTran (Batchnorm/ReLU)	(3,2)	(2,2)	(1,1)	(128, 51, 26)
ConvTran (Batchnorm/ReLU)	(3,3)	(2,2)	(1,1)	(64, 101, 51)
ConvTran (Tanh)	(2,2)	(2,2)	(1,1)	(1, 200, 100)
Encoder				
Input	–	–	–	(1, 200, 100)
Conv (LeakyReLU/Batchnorm)	(2,2)	(2,2)	(1,1)	(64, 101, 51)
Conv (LeakyReLU/Batchnorm	(3,3)	(2,2)	(1,1)	(128, 51, 26)
Conv (LeakyReLU/Batchnorm	(3,2)	(2,2)	(1,1)	(256, 26, 14)
Conv (LeakyReLU/Batchnorm	(2,2)	(2,2)	(1,1)	(512, 14, 8)
Conv (LeakyReLU/Batchnorm	(2,2)	(2,2)	(1,1)	(1024, 8, 5)
Conv (LeakyReLU/Batchnorm	(2,2)	(1,1)	(0,0)	(2048, 7, 4)
Conv (LeakyReLU/Batchnorm	(2,2)	(1,1)	(0,0)	(4096, 6, 3)
Conv	(6,3)	(1,1)	(0,0)	(4, 1, 1)
Discriminator				
Input	–	–	–	(1, 200, 100)
Conv (LeakyReLU)	(100,50)	(14,4)	(1,1)	(64, 8, 14)
Conv (Sigmoid)	(8,14)	(1,1)	(0,0)	(1, 1, 1)

	Generator	Encoder	Discriminator
Optimizer	Adam	Adam	Adam
(β_1,β_2)	(0.500,0.500)	(0.500,0.500)	(0.500,0.999)
Learning rate	0.002	0.002	0.0002

β = Momentum

Table 4 Architecture of the ensemble transfer learning networks

	Layer 1	Layer 2	Layer 3	Layer 4	Layer 5	Layer 6	Learning rate	Optimizer	Epoch
NN_1	5	20	200	100	50	2	0.01	Adam	100
NN_2	5	20	200	100	50	2	0.01	Adam	100
.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.
NN_{200}	5	20	200	100	50	2	0.01	Adam	100

As outlined in "Experiments setup" section, to assess the effectiveness of the proposed EGAN approach in predicting milling stability across various target conditions, three different tool setups and two distinct target stability maps for prediction within each setup are considered. To ensure a comprehensive assessment of the proposed approach's performance and facilitate the creation of mean and confidence intervals(CIs),thegridsearchapproachisemployedtoselect 10 distinct sets of five test cuts. In addition, to conduct a comparative analysis against state-of-the-art techniques, two evaluation metrics, the Geometric Mean (G-Mean) Espindola and Ebecken (2005) and L_1 norm Sinwar and Kaushik (2014), are employed. The G-Mean assesses the measures the extent of dissimilarity between the predicted and actual stability maps by computing the absolute differences between correspondingelementsandsummingthemup.Theseevaluationmetrics are chosen to provide a comprehensive evaluation of the proposed approach's performance and assess its practicality in real-world applications.

Table 5 presents the average performance metrics and their 95% confidence intervals (CIs) over 10 different sets of five and 10 test cuts for both the proposed EGAN approach and the benchmark methods. The evaluation is performed on Targets 1 and 2 for each of the three tool setups. These results provide a comprehensive assessment of the proposed approach's performance in predicting

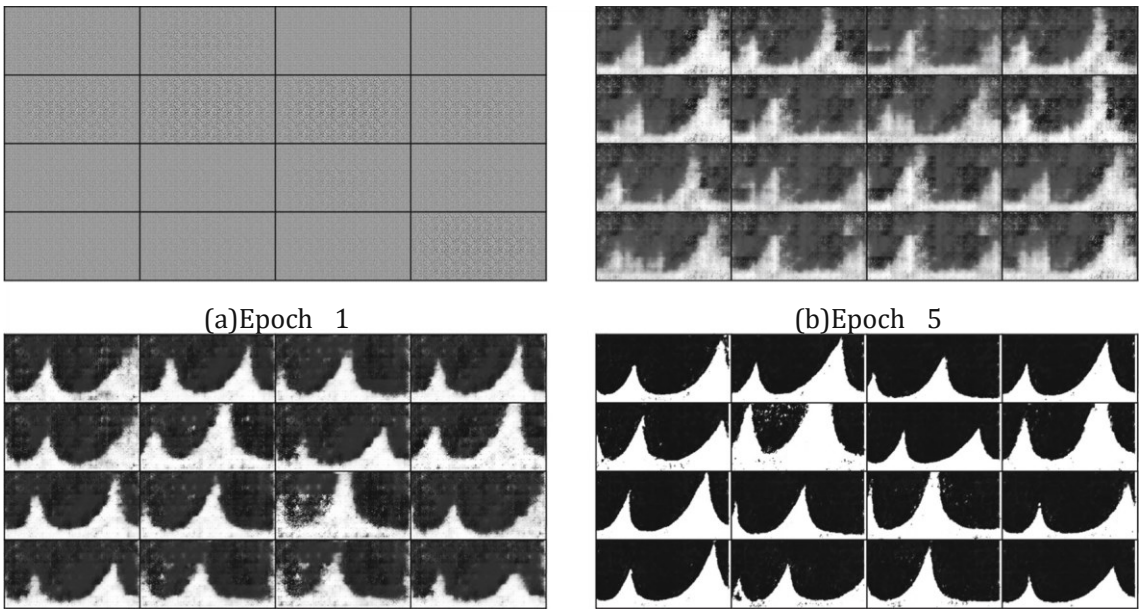
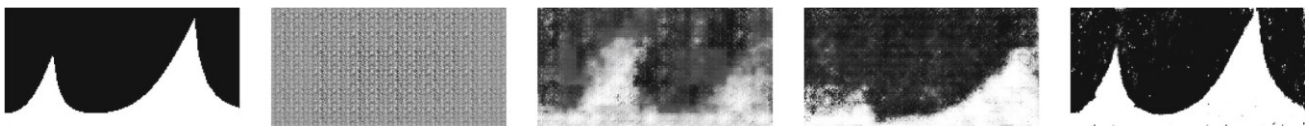


Fig. 14 Stability maps produced by the generator during the training process for Setup 2



(a) Input stability (b) Epoch 1 (c) Epoch 5 (d) Epoch 10 (e) Epoch 100 map

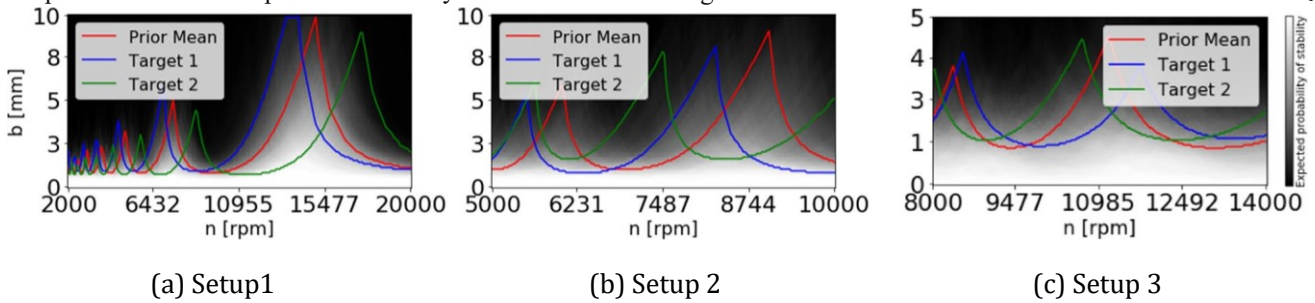
Fig. 15 The reconstruction ($G(E(x))$) of the "Prior Mean" stability map in Setup 2 during the training process

approach's ability to balance false positives (incorrectly labeling stable cuts as unstable) and false negatives (incorrectly labeling unstable cuts as stable). The G-Mean computes the square root of the product of sensitivity and specificity. Sensitivity represents the proportion of true positives to the total number of actual positives, while specificity represents the proportion of true negatives to the total number of actual negatives. Meanwhile, the L_1 norm

milling stability for various tool setups and target conditions. Additionally, to visually showcase the stability predictions for each of the scenarios, Figs.18, 19, 20, 21, 22 and 23 present three representative samples from the 10 sets of test cuts performed for each scenario, namely Setup 1 Target 1, Setup 1 Target 2, Setup 2 Target 1, Setup 2 Target 2, Setup 3 Target 1, and Setup 3 Target 2, respectively. These figures provide a visual representation of the stability

predictions made by the EGAN approach and the benchmark methods, which further complement the quantitative results presented in Table 5. Specifically, each figure displays the predicted probabilistic stability maps generated by the EGAN and benchmark approaches for each of the target within different setups. The true stability limit for each target is shown as a red line in all figures. These figures also show the predicted stability boundary, which is determined by applying a threshold of 0.5 to the output probability of the approaches. These figures enable a direct comparison between the predicted stability boundaries and the true stability limit, thereby providing insight into the accuracy and effectiveness of the EGAN and benchmark approaches in predicting milling stability for each scenario. In addition, Figs. 29, 30, and 31 in the appendix present the test cuts selected by the grid search approach for these numerical experiments.

The comparison between the EGAN and Bayesian learning approaches is presented in Table 5. As can be seen, the EGAN approach outperforms the Bayesian learning approach in predicting stability limits, as indicated by the higher values of the G-Mean and/or lower values for L_1 metrics under both five and 10 test cuts in all scenar-



(a) Setup1 (b) Setup 2 (c) Setup 3

Fig. 16 Prior probabilistic stability map for each setup obtained by 200 sample stability maps that are used to pre-train neural networks in ensemble transfer learning

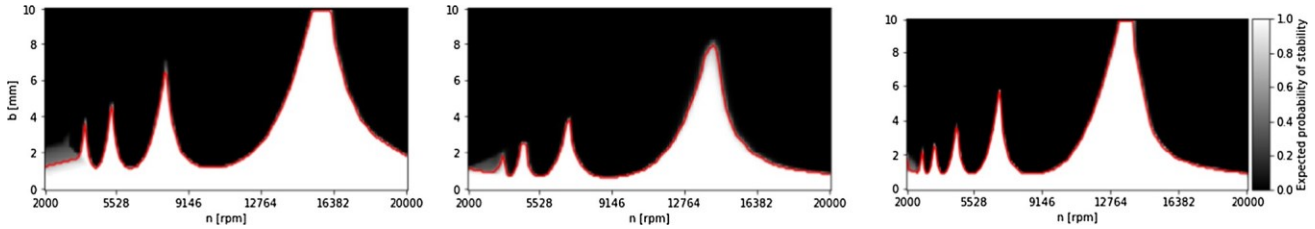


Fig. 17 Three sample stability maps from the training dataset of Setup 1 represented as red lines, along with their respective output probability of stability predictions obtained through the process of transfer learning

Table 5 Mean and 95% CI of the performance metrics over 10 different sets of five and 10 test cuts for the Bayesian learning, ensemble transfer learning, and EGAN approaches

			5 Test Cuts					10 Test Cuts		
			BL	ETL	EGAN			BL	ETL	EGAN
Setup 1	Target 1	G-Mean	75.90 ± 0.55	83.05 ± 2.83	84.36 ± 4.22			78.66 ± 0.99	88.55 ± 2.76	87.94 ± 3.62
		L_1	0.31 ± 0.01	0.14 ± 0.01	0.01	0.14 ± 0.03	0.27	0.27 ± 0.01	0.10 ± 0.01	0.11 ± 0.02
	Target 2	G-Mean	76.68 ± 1.01	83.59 ± 3.09		**88.37*** ± 3.37	79.16	79.16 ± 1.75	87.25 ± 2.62	**91.77*** ± 5.46
		L_1	0.32 ± 0.01	0.14 ± 0.02		**0.11*** ± 0.02	0.28	0.28 ± 0.02	0.09 ± 0.02	0.09 ± 0.03
Setup 2	Target 1	G-Mean	79.91 ± 1.09	81.01 ± 3.30	83.22 ± 6.15			83.24 ± 0.86	88.76 ± 3.08	87.21 ± 5.58

		L_1	0.30 ± 0.01	0.17 ± 0.02	0.16 ± 0.04	0.25 ± 0.01	0.11 ± 0.01	0.13 ± 0.03
		G-Mean	84.14 ± 0.93	77.39 ± 4.63	**82.14*** ± 3.55	85.94 ± 0.68	84.71 ± 3.46	82.67 ± 2.58
	Target 2	L_1	0.26 ± 0.01	0.19 ± 0.02	**0.16*** ± 0.02	0.23 ± 0.01	0.14 ± 0.02	0.15 ± 0.02
		G-Mean	86.19 ± 0.55	87.67 ± 1.5	**90.36*** ± 2.90	88.59 ± 0.56	90.69 ± 2.66	**92.02*** ±
Setup 3	Target 1	L_1	0.24 ± 0.01	0.16 ± 0.01	**0.12*** ± 0.02	0.19 ± 0.01	0.12 ± 0.01	1.15 **0.10*** ±
	Target 2	G-Mean	84.74 ± 0.43	87.34 ± 0.75	87.43 ± 2.15	87.55 ± 0.74	90.97 ± 2.51	0.01
		L_1	0.25 ± 0.01	0.16 ± 0.01	**0.14*** ± 0.02	0.20 ± 0.01	0.11 ± 0.01	**93.57*** ± 2.01
								0.08* ± 0.02

The bold formatting is used to highlight instances where EGAN demonstrates superior performance compared to benchmarks *BL* Bayesian learning, *ETL* ensemble transfer learning

$^*p$ value < 0.1: Paired *t*-tests between the EGAN and ensemble transfer learning

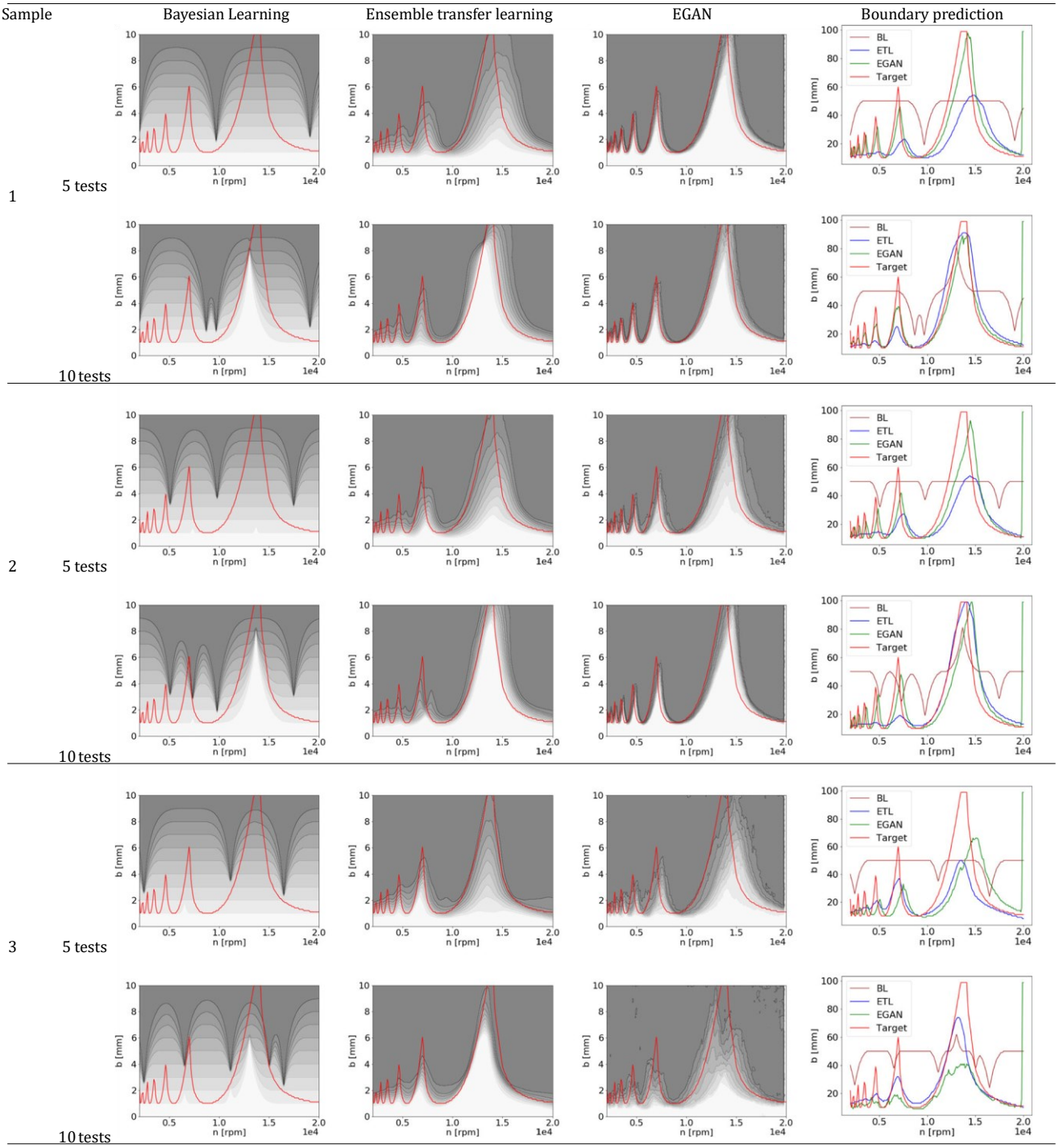


Fig.18 ThenumericalresultsoftheBayesianlearning,ensembletrans- fer learning, and EGAN approaches in predicting Target 1 within setup 1 1 using five and ten test cuts. The results are shown for three representa- and 2 tivesamplesfromthe10setsoftestcuts.Asdemonstratedbytheresults, the EGAN approach outperforms benchmarks especially in samples

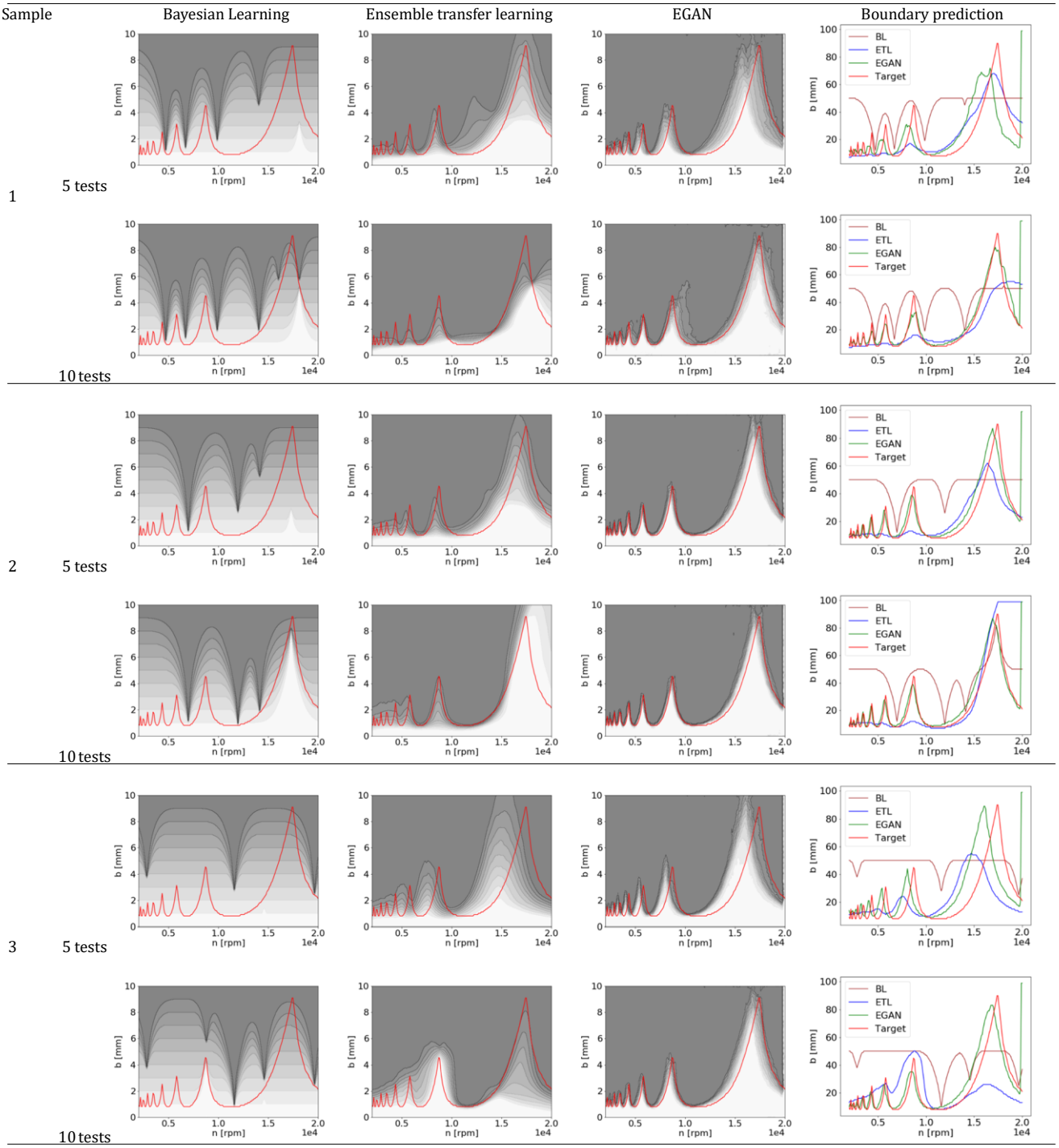


Fig.19 ThenumericalresultsoftheBayesianlearning,ensembletrans- sentativesamplesfromthe10setsoftestcuts.Asdemonstratedbythe fer learning, and EGAN approaches in predicting Target 2 within setup results, the EGAN approach consistently outperforms the benchmarks 1 using five and ten test cuts. The results are shown for three repre- across all samples, particularly in scenarios involving ten test cuts

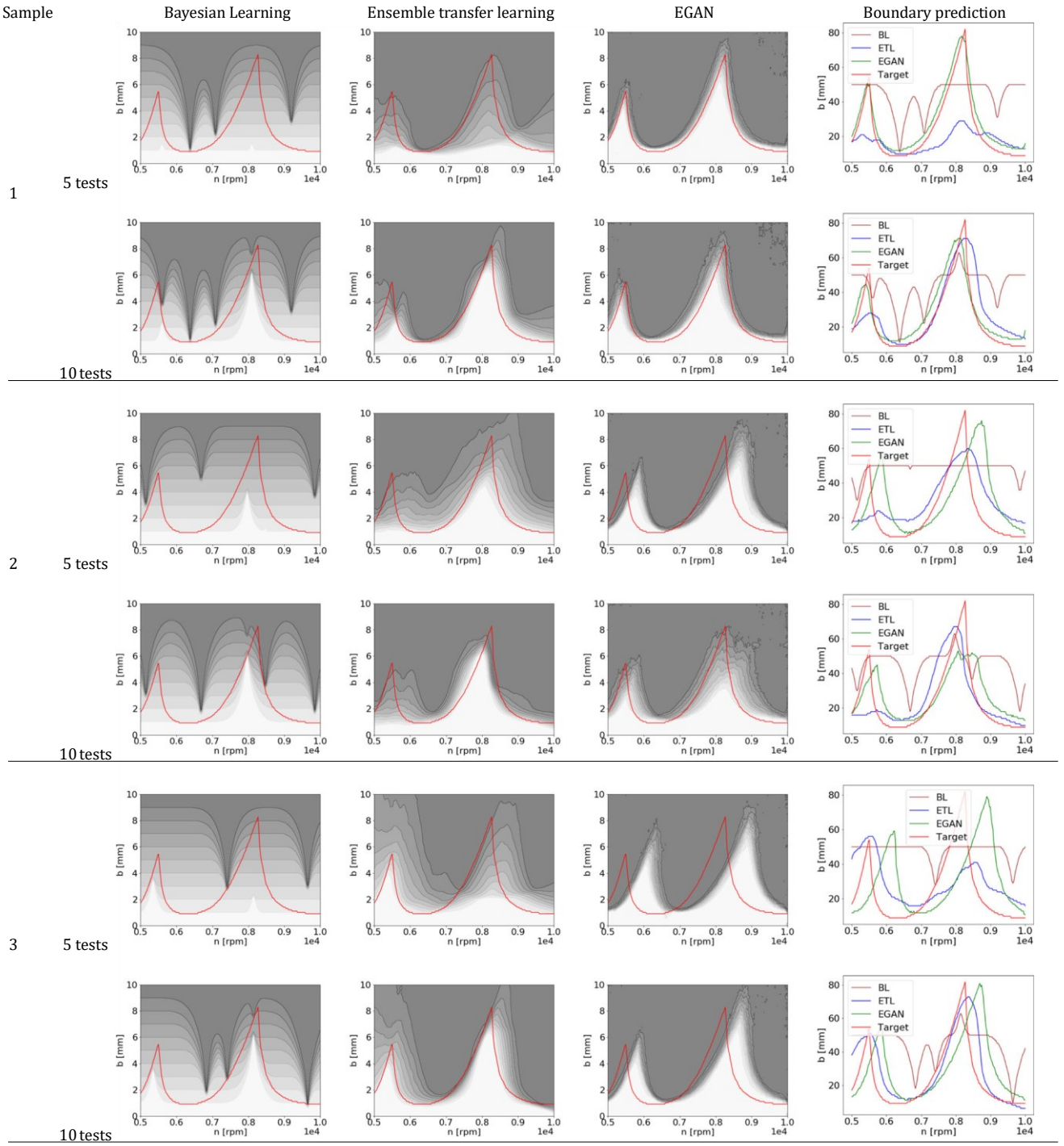


Fig.20 ThenumericalresultsoftheBayesianlearning,ensembletransfer learning, and EGAN approaches in predicting Target 1 within setup 2 using five and ten test cuts. The results are shown for three representative samples from the 10 sets of test cuts. As demonstrated by the results, the EGAN approach exhibits superior performance compared to the benchmarks in sample 1. However, it is notable that none of the approaches achieved satisfactory predictions for the other two test samples

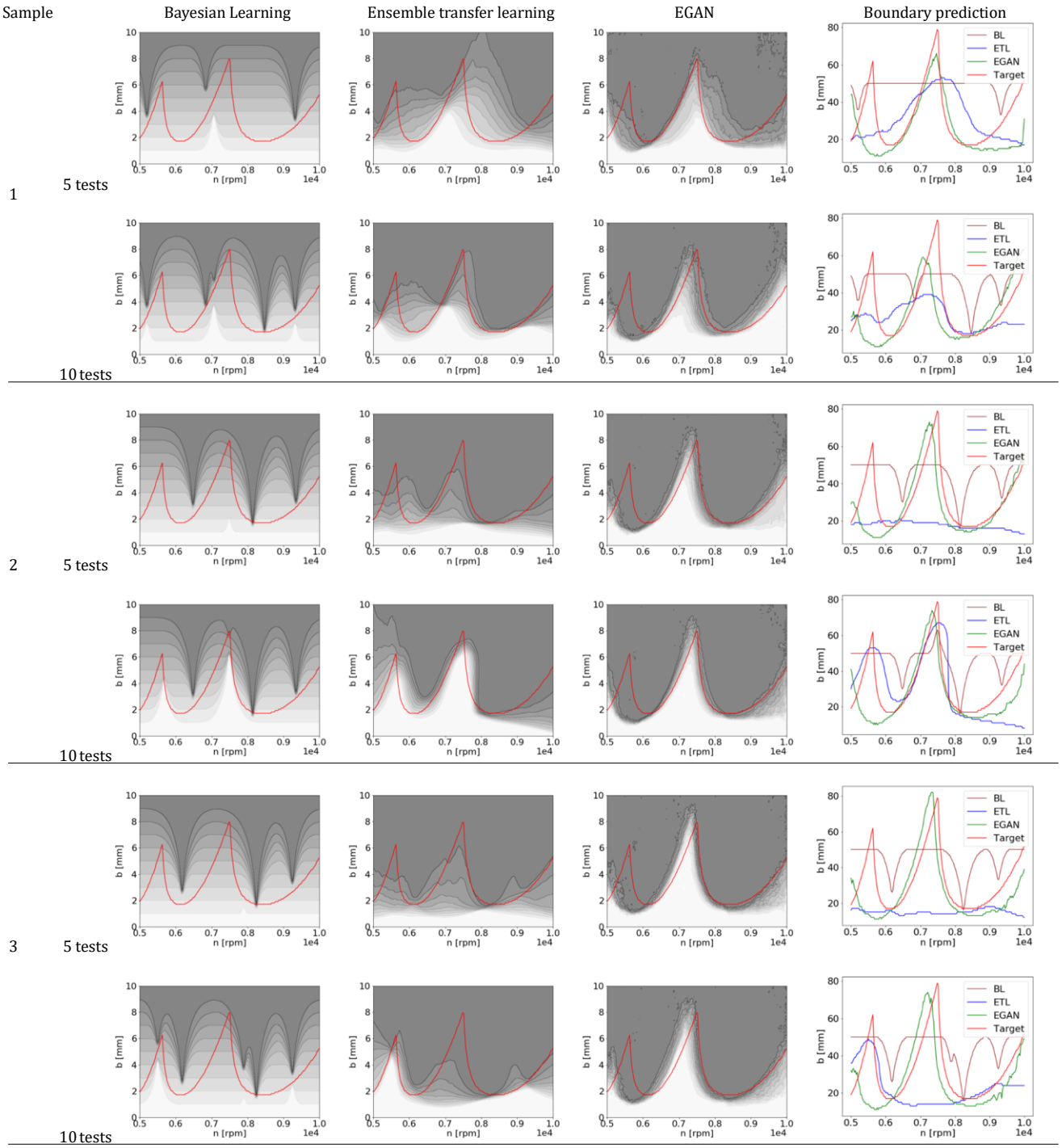


Fig.21 ThenumericalresultsoftheBayesianlearning,ensembletrans- sentativesamplesfromthe10setsoftestcuts.Asdemonstratedbythe fer learning, and EGAN approaches in predicting Target 2 within setup results, the EGAN approach consistently outperforms the benchmarks 2 using five and ten test cuts. The results are shown for three repre- across all samples

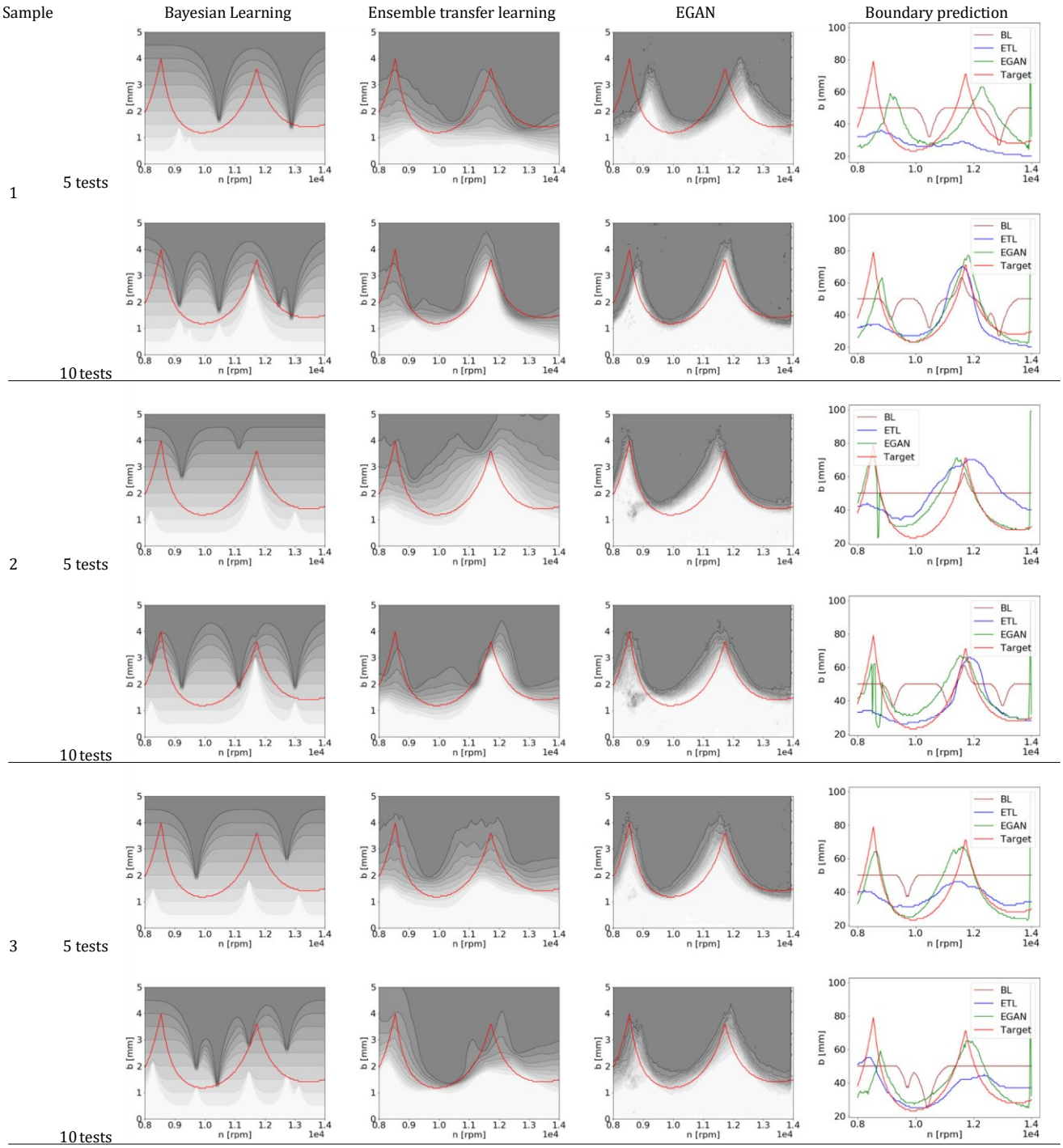


Fig.22

ThenumericalresultsoftheBayesianlearning,ensembletransferlearning, andEGANapproachesinpredictingTarget1withinsetup3 usingfiveandtentestcuts.Theresultsareshownforthreerepresentative samples from the 10 sets of test cuts. As demonstrated by the results, the EGAN approach consistently demonstrates superior performance over the benchmarks across a majority of the samples particularly in samples 1 and 3

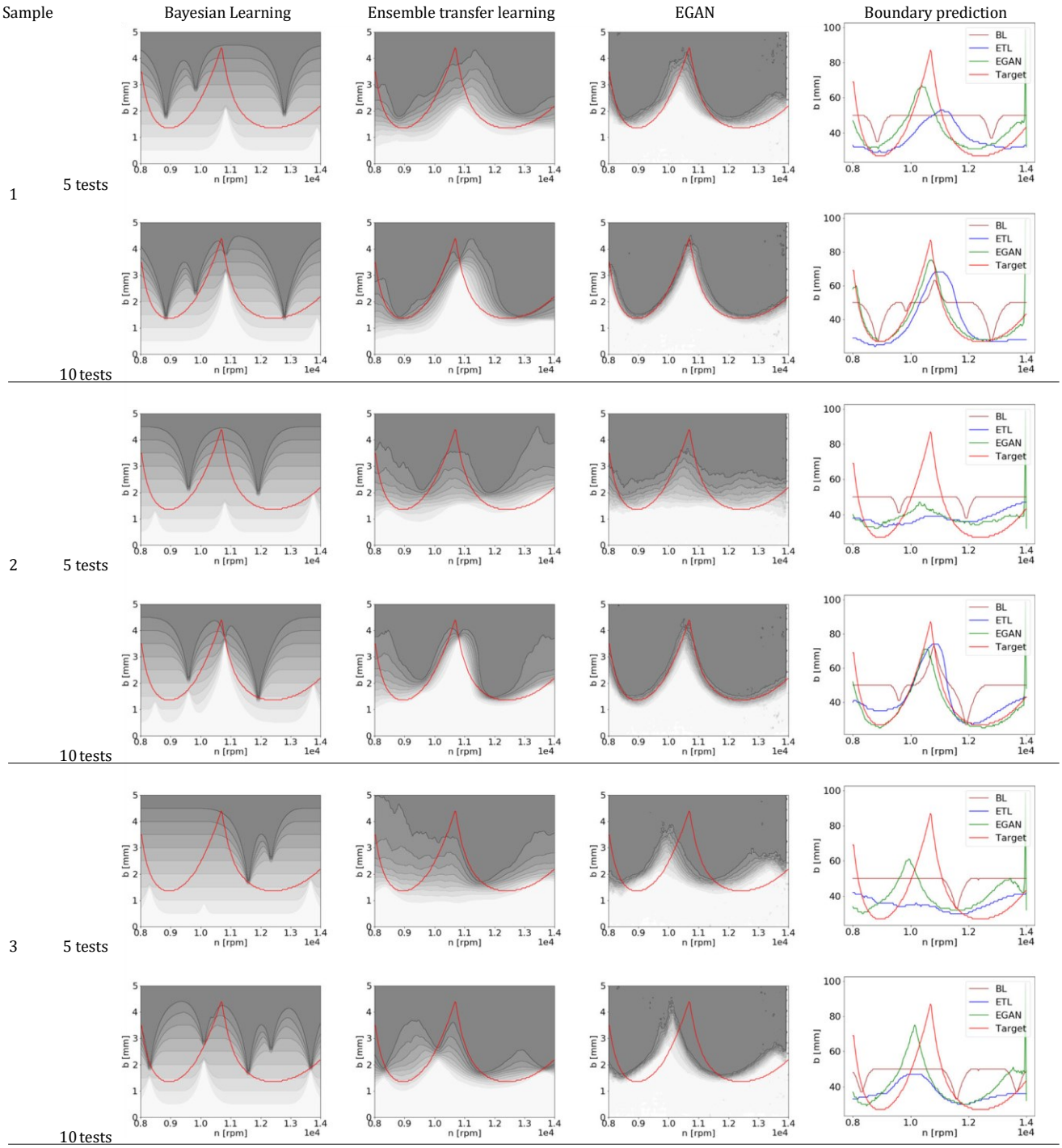


Fig.23 ThenumericalresultsoftheBayesianlearning,ensembletransfer learning, and EGAN approaches in predicting Target 2 within setup 3 using five and ten test cuts. The results are shown for three representative samples from the 10 sets of test cuts. As demonstrated by the results, the EGAN approach demonstrates superior performance in comparison to the benchmarks in sample 1. While in samples 2 and 3, none of the approaches perform adequately with five test cuts; however, it is notable that EGAN outperforms the other approaches with ten test cuts

ios. Paired *t*-tests confirm the statistical significance of this improvement. The only exception to this trend is observed in predicting Target 2 in setup 2, where the Bayesian learning approach appears to outperform the EGAN approach in terms of the G-Mean metric measurements (84.14 ± 0.93 versus 82.14 ± 3.55 under 5 test cuts, and 85.94 ± 0.68 versus 82.67 ± 2.58 under 10 test cuts). However, considering the L_1 norm, it is the EGAN approach that outperforms the Bayesian learning approach (0.26±0.01 versus 0.16±0.02 under 5 test cuts, and 0.23 ± 0.01 versus 0.15 ± 0.02 under 10 test cuts). The sample results presented in Fig.21 confirm the superiority of the EGAN approach to the Bayesian learning in predicting stability limits as indicated by the L_1 norm results.

Table 5 also provides a comprehensive comparison of the EGAN and ensemble transfer learning approaches. The results reveal that the EGAN approach exhibits superior performance to the ensemble transfer learning approach for most scenarios, based on G-Mean and/or L_1 metrics, for five test cuts. The statistical significance of this improvement is confirmed through paired *t*-tests, as presented in the table. For instance, in Target 1 of Setup 3, the EGAN approach attains a G-Mean measurement of 90.36, which is higher than the G-Mean measurement of 87.67 achieved by the ensemble transfer learning approach. Similarly, the L_1 norm measurements for the EGAN and ensemble transfer learning approaches in this scenario are 0.12 and 0.16, respectively, indicating a reduction of 0.04 in the L_1 norm bytheEGANapproach.However,thereareexceptionstothis trend,asobservedinTarget1ofSetup1andTarget1ofSetup 2, where both approaches exhibit similar performance under both metrics. For instance, the G-Mean measurement for EGAN in Setup 1 Target 1 is 84.36±4.22, while the G-Mean measurement for ensemble transfer learning is 83.05 ± 2.83 (paired *t*-tests > 0.1). Furthermore, the results indicate that increasing the number of test cuts to 10 enhances the performance of all approaches. Nonetheless, for 10 test cuts, the EGAN approach still outperforms the ensemble transfer learning in predicting Target 1 in Setup 2, as well as Targets 1 and 2 in Setup 3. In the remaining setups, no significant differences are observed between these two approaches.

The results presented in Table 5 and Figs.18, 19, 20, 21, 22 and 23 highlight the effectiveness of the proposed EGAN approach in accurately identifying the actual stability boundary, even with a limited number of numerical tests (to present cutting experiments) employing only five or 10 test cuts. For instance, the third sample presented in Fig.21 demonstrates the ability of the EGAN approach to detect the target stability boundary using only five or 10 test cuts, whereas the ensemble transfer learning approach fails to identify the boundary evenwith10tests.Thisexemplifies thepotentialof the EGAN approach to achieve accurate stability predictions in scenarios with limited experimental data. Furthermore, Figs.18, 19, 20, 21, 22 and 23 demonstrate the ability of the EGAN approach to predict all stability maps, whereas the ensemble transfer learning approach fails to do so. For instance, the samples presented in Fig.19 illustrate that the EGANapproachcanpredicttheentirestabilityboundary,not only the high spindle speed range, with only limited information from five or 10 tests. Conversely, in most scenarios, the ensemble transfer learning approach is unable to predict theseboundariesaccurately.Thisobservationemphasizesthe superiority of the EGAN approach over the ensemble transfer learning approach in predicting all stability maps with limited experimental data.

However, it is important to note that there are some cases where the EGAN approach does not receive sufficient information to predict the stability boundary with only five tests. For instance, the second sample in Fig.23 demonstrates this limitation. As seen in the figure, the information obtained from five test cuts is insufficient, resulting in failure for both the EGAN and ensemble transfer learning approaches to find the stability boundary. Nevertheless, by increasing the number of tests to 10, both approaches are able to predict the stability boundary with good accuracy. Variations in the shape and positioning of predicted stability boundaries have emerged as a noteworthy consideration, influenced by the selection of test cuts. These variations signify the sensitivity of the EGAN approach to the specificities of the chosen test cases. Overall, the results of this study demonstrate the potential of the EGAN approach as an effective method for stability prediction in scenarios with limited experimental data, and highlight its superiority over the ensemble transfer learning approach in accurately identifying the stability boundary.

Experimental validation

In this section, a series of experiments are conducted to further verify the proposed EGAN approach using real-world cutting data. The tool setup 3 and a new tool setup, named tool setup 4, are used for this evaluation.

Actual experiment for setup 3

In this section, actual cutting tests are performed for tool setup3,asoutlinedinTables1and2,toutilizetheEGANand ensemble transfer learning approaches to predict the actual (unknown) stability map that best represents the test cuts. To make predictions, the previously trained networks in EGAN and ensemble transfer learning are utilized to provide a subset of the performed test cuts as input and output the actual stability map. This approach is taken to further demonstrate the effectiveness of the EGAN approach in practical settings.

Fig. 24 Machining setup for cutting tests Schmitz et al. (2022)

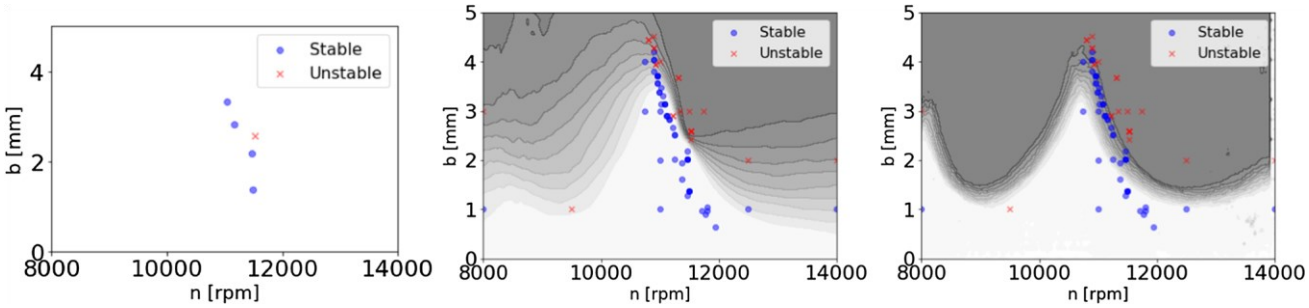
These test cuts are performed on a DMG Mori Ultrasonic 65 machining center using a three flute 12.7 mm diameter carbide endmill (Robbjack FMHV-304-16). The work material is 7050-T7451 aluminum. The test setup is shown in Fig.24. Figure25 displays the selected five test cuts and the predicted probabilistic stability map generated by the ensemble transfer learning, and EGAN approaches. The accuracy and efficacy of these predictions can be evaluated by examining the remaining actual cutting tests that are displayed in the figure alongside the predictions. As can be observed from Fig.25, the proposed EGAN approach is capable of generating a stability map that most accurately reflects the actual cutting tests. While it is not possible to carry out a comprehensive comparison between these approaches due to the limited number of available test cuts, comparing the probabilistic maps shows that the ensemble transfer learning approach is uncertain about the stability boundary in areas where no information is available, while the EGAN can predict the boundary even in these areas.

Actual experiment for setup 4

TofurtherinvestigatetheeffectivenessoftheEGANapproach on actual experiments, a new tool setup (Setup 4) is introduced for a climb-milling process. This tool setup includes a cutting tool with a diameter of 12.7 mm, equipped with four cutting teeth. The recommended feed rate for this tool is 0.1mm per tooth. Additionally, the radial immersion is set at 40%. For the uncertain parameters, three distributions are considered, referred to as Case 1, Case 2, and Case 3, where somevaluesareoverestimated,someareunderestimated,and some provide a good initial estimate, with the largest uncertainty generally applied to damping ratio. Specifically, Case 1representsasituationwherethetoolFRFwasmeasured,but the FRF changes with spindle speed. The cutting force model is based on prior experience using similar tool/workpiece material combinations, but is not measured specifically for this case. Case 2 represents a situation where the FRF uncertainty is low, but not zero due to modeling efforts. The work material is similar to previous testing, but not the same. Case 3 represents a wide distribution where very little is known about the system to explore the bounds of the EGAN capabilities.

Tables 6, 7 and 8 contain detailed information about these distributions. These tables also provide the nominal values for the unknown parameters obtained through tap testing. In Figure 26, the red line identifies the stability boundary for these nominal values, along with the test cuts performed. As seen in this figure, there is uncertainty in the analytical predictionand,consequently,thenominalstabilitymapisnot able to accurately predict all test cuts. The goal is to assess the prediction performance of the EGAN approach for either the nominal or actual stability map.

The EGAN and ensemble transfer learning approaches are trained on stability maps generated by each distribution presented in Tables 6, 7 and 8. To evaluate the performance of these models, a set of five test cuts is selected and used



(a) Test cuts (b) Ensemble transfer learning (c) EGAN

Fig. 25 The actual cutting tests in Setup 3 and predicted probabilistic stability maps generated by the ensemble transfer learning and EGAN approaches using five test points

Table 6 Case 1: Tight normal distribution for the Setup 4

Parameter	Nominal value	Mean (μ)	StdDev (σ)	Error (stdDevs)	Error (%)
f_n (Hz)	1998	1950	100	−0.48	−0.02
$k(N/m)$	4.47×10^6	5×10^6	1×10^6	0.53	0.12
ζ	0.0123	0.01	0.0025	−0.92	−0.19
$K_s(N/m^2)$	6×10^8	6×10^8	1×10^8	0	0
β (degree)	68	65	5	−0.6	−0.04

Table 7 Case 2: Wide normal distribution for the Setup 4

Parameter	Nominal value	Mean (μ)	StdDev (σ)	Error (stdDevs)	Error (%)
f_n (Hz)	1998	1800	100	−1.98	−0.1
$k(N/m)$	4.47×10^6	5×10^6	2×10^6	0.265	0.12
ζ	0.0123	0.01	0.01	−0.23	−0.19
$K_s(N/m)$	6×10^8	7×10^8	1×10^8	1	0.17
β (degree)	68	60	10	−0.8	−0.12

Table 8 Case 3: Wide uniform distribution for the Setup 4

Parameter	Nominal value	Min	Max	Width as % of nominal
f_n (Hz)	1998	1600	2400	0.4
$k(N/m)$	4.47×10^6	3×10^6	6×10^6	0.67
ζ	0.0123	0.005	0.025	1.63
$K_s(N/m^2)$	6×10^8	3×10^8	10×10^8	1.17
β (degree)	68	55	75	0.29

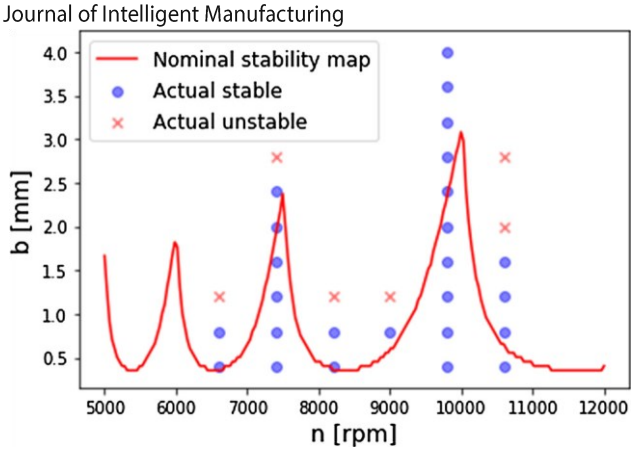


Fig. 26 Actual test cuts along with the stability boundary for the nominal values obtained through tap test in Setup 4

as input to predict the stability maps. Figure27 displays the selected test cuts alongside the probabilistic stability predictionsofthemodelsandtheexperimentalresultsforeachcase. The predicted stability boundary is also shown in the figure as a green line, which is determined by applying the threshold 0.5 on the probabilistic maps. The results indicate that, while the EGAN approach cannot predict the true stability map covering all cutting tests, it can identify the best stability map based on the available information. In Case 1, the tight normal distribution limits the EGAN's ability to learn various stability behaviors including the true stability, and thus, the best stability map that it finds closely matches the nominal map. Conversely, in Cases 2 and 3, where a wider range of stability behavior is observed, the EGAN can adjust the nominal stability boundary and predict a stability map that covers more information than the nominal map. The figure illustrates that this adjustment is more significant when the EGAN is trained on a wide uniform distribution. Notably, the EGANoutperformstheensembletransferlearningapproach, particularly in Cases 2 and 3, where the latter exhibits low accuracy in predicting the true stability boundary.

This example experiences a degree of mode collapse. For instance, inthe case of a wide normal distribution, the EGAN encounters various stability maps but fails to identify the true stability map. This could be due to the fact that the EGAN doesnotlearnandgenerateawiderangeofstabilitybehaviors as observed in the training dataset. One possible explanation for this is the imbalanced dataset, where some similar

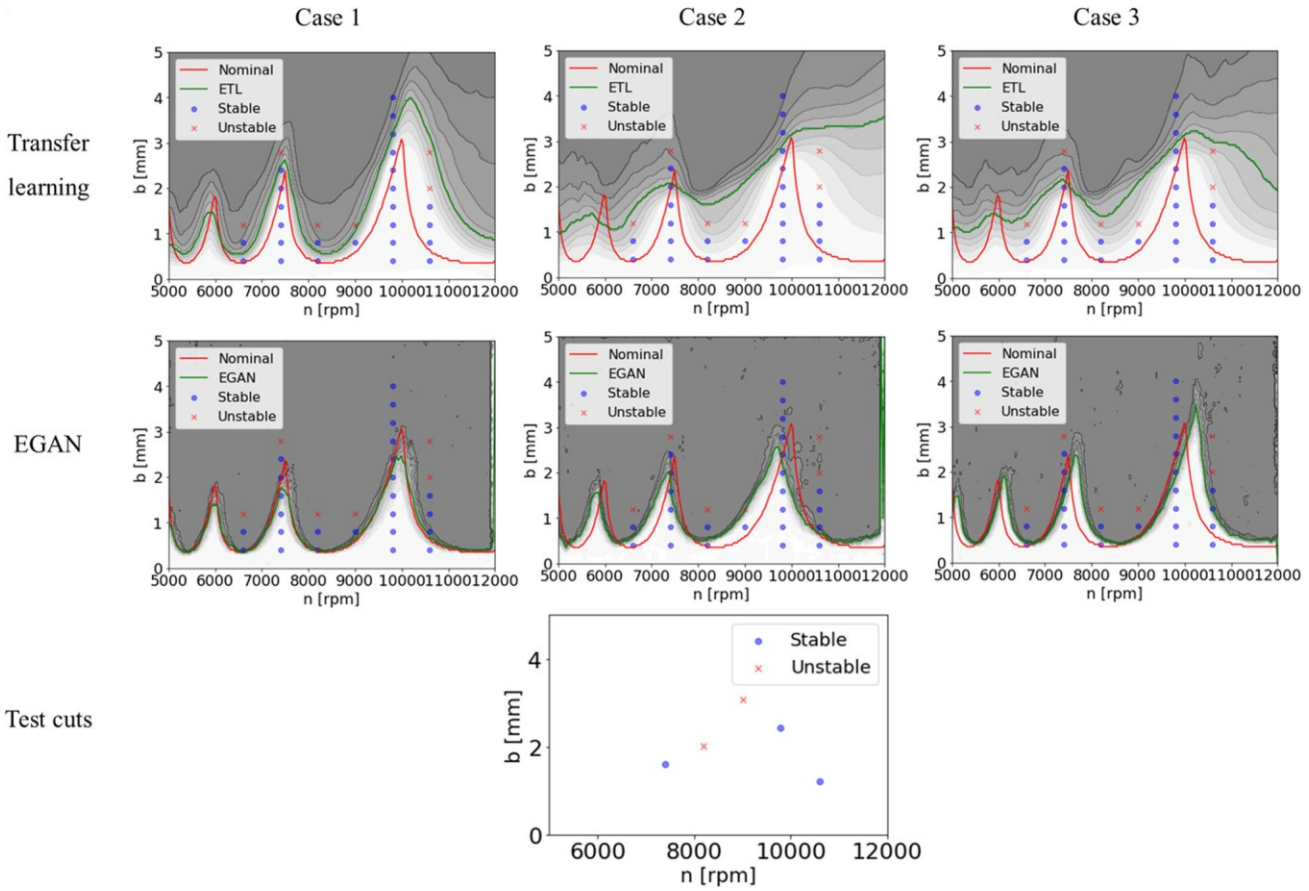


Fig. 27 The performance of the EGAN and ensemble transfer learning approaches on actual experiments in Setup 4, using Case 1, Case 2, and Case 3. The figure shows the predictions of the models compared to the experimental results and nominal stability map

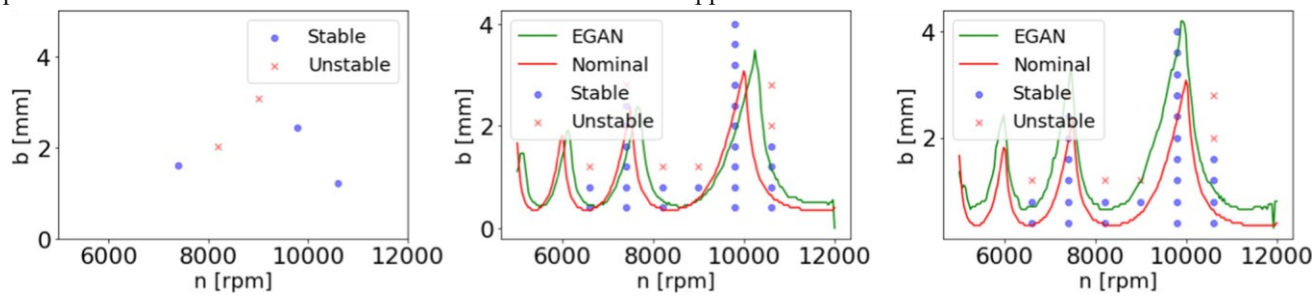
samples occur more frequently than others. To address this, K-Means is used to cluster the stability maps and up-sample the clusters with fewer samples to balance the dataset. This approach prevents the EGAN from being biased towards the clusters with more samples. The performance of the EGAN approach in predicting the boundary is depicted in Fig.28 before and after applying K-Means to balance the dataset generated under Case 3 in Setup 4. The figure demonstrates a substantialimprovementintheEGAN'sperformancefollowing the dataset balancing. The stability map that is calculated using tap-testing is inaccurate. Even so, EGAN is still able to find a stability map that's able to match the actual cutting test results. The results suggest that addressing the problems of instability of training and mode collapse can significantly enhancetheEGANapproach'sperformance.Thestudyanticipates that the rapid development of GANs will soon yield a robust structure that does not have these issues.

It is important to consider that variations in workpiece material have a direct impact on cutting force coefficients. This implies that altering the work material could lead to changes in the resulting stability map. However, in the context of the proposed EGAN approach, our training process relies on a dataset of stability maps derived from a physicsbased analytical stability model. This dataset accounts for a distribution of unknown parameters, encompassing cutting force coefficients among others. Consequently, it encompassesstabilitymapsrepresentingaspectrumofcuttingforce coefficient variations.

The training of the generator and encoder networks within the EGAN framework is rooted in this diverse dataset. This enables the model to acquire insights into the inherent stability behaviors and patterns linked to different combinations of materials and machining parameters. Given that the model is trained across a range of distinct cutting

forces, it exhibits an inherent adaptability that may extend to different work materials within the bounds of its training data. However, it is worth noting that to ensure the seamless adaptation of the EGAN approach to new materials, retraining the model using stability maps generated from the dynamic characteristics of the new material is a prudent recommendation. This ensures that the model captures the material-specific behavior required for accurate stability predictions.

The novelty of the approach lies in the use of EGAN and the regularization property of the latent space to predict stability maps with limited information, which has the potential to reduce the number of cutting tests needed to predict stability maps accurately. The study demonstrates the effectiveness of the proposed EGAN approach through extensivenumericalsimulationsandreal-worldexperiments, with a comprehensive comparison against state-of-the-art approaches.



(a) Test cuts (b) EGAN without K-Means (c) EGAN with K-Means

Fig. 28 The performance of the EGAN with and without applying K-Means to balance the training dataset generated under Case 3 in Setup 4

Conclusion and future work

In this study, Encoder GAN (EGAN) is introduced for predictingstabilitymapsusingsimulatedandexperimentaldata. The proposed approach is based on Generative Adversarial Networks (GANs) and consists of the generator, encoder, and discriminator subnetworks. The simulated data generatedusingaphysics-basedanalyticalstabilitymodelareused as training dataset for EGAN. The generator learns to mimic the physics-based model's behavior by outputting a plausible stability map. The encoder learns to project stability maps back into the corresponding latent space parameter, while the discriminator distinguishes generated/fake stability maps from sample maps in the input dataset. The EGAN approach is designed to generate plausible and distinctive stability maps for each latent space parameter set, while simultaneously enabling the encoder to identify the unique set of latent parameters associated with each stability map. This is achieved through a rigorous training process that involves regularization of the latent space, ensuring that similar stability maps are assigned to nearby points in the latent space. These properties allow the EGAN to accurately reconstruct partialstabilityinformationandeffectivelyidentifytheactual stability map. To obtain partial information about the target stabilitymap,alimitednumberofcuttingtestsareinputintoa non-physics-based Bayesian heuristics approach. Once partial information is obtained, the trained EGAN is utilized to predict the actual stability map.

Several promising directions for future research are suggested. For instance, the proposed approach could be evaluatedusingothervariationsofGANs,suchasvariationalGAN, cycle GAN, DRAGAN, and others, to reduce mode collapse and instability of training problems. These methods may further enhance the accuracy and robustness of the proposed approach. Moreover, future studies can explore alternative approaches to the grid search method proposed in this study, whichcouldprovideadditionalinformationtotheEGANand potentially accelerate the identification of the target stability map. Such investigations may improve the efficiency and effectiveness of the proposed approach and contribute to its broader applicability. Furthermore, the effectiveness of the proposedapproachcanbeevaluatedforothermachiningprocesses, such as turning. Investigating the applicability of the proposed method to other machining processes would provide valuable insights into its generalizability and potential for wider adoption across various industrial applications.

Acknowledgements The research was partially supported by the DepartmentofEnergy(DOE),AdvancedManufacturingOffice(AMO), Award Number: DE-EE0009400. The authors would also like to acknowledge support from the NSF Engineering Research Center for Hybrid Autonomous Manufacturing Moving from Evolution to Revolution (ERC-HAMMER) under Award Number EEC-2133630.

Data availability The data that support the findings of this study are available at https://github.com/srezaei90/GANs-to-predict-stabilitymaps-in-milling-machining.git.

Declarations

Conflict of interest No potential conflict of interest was reported by the authors.

Appendix

See Figs. 29, 30 and 31.

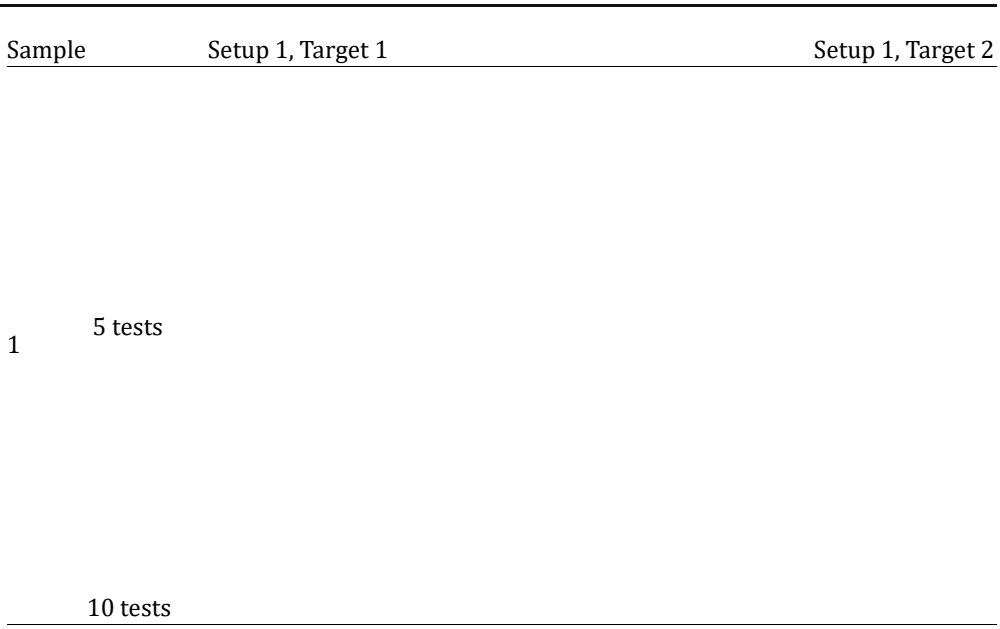
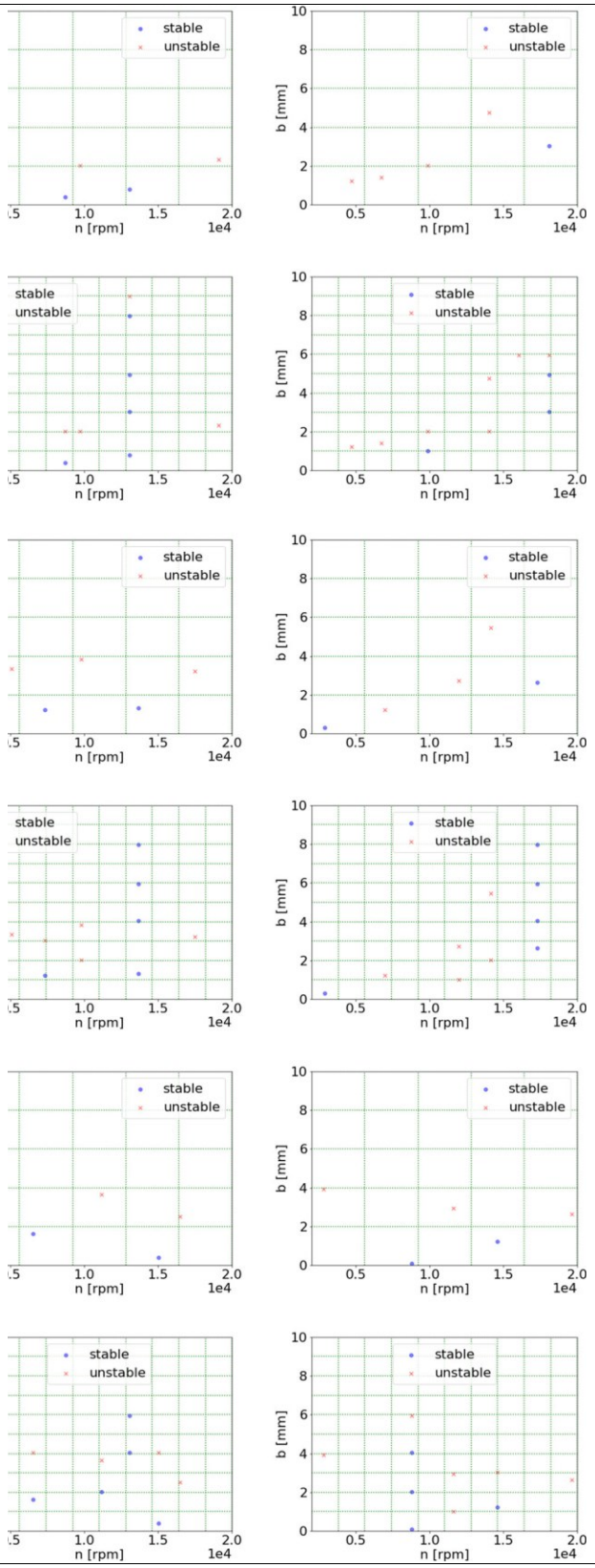


Fig. 29 Sample test points for numerical experiments: Setup 1

2



5 tests

10 tests

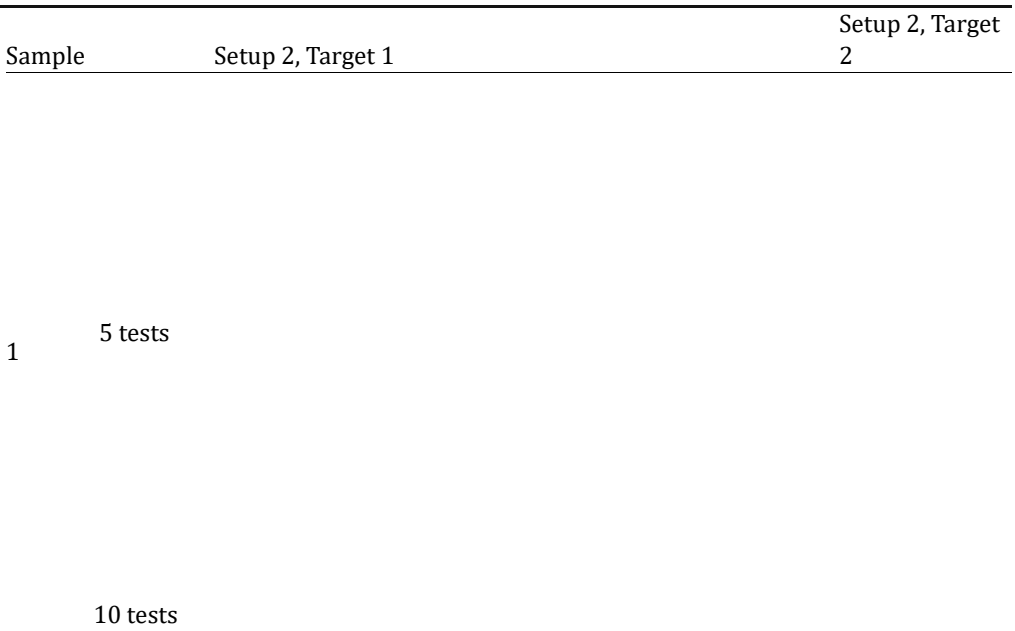
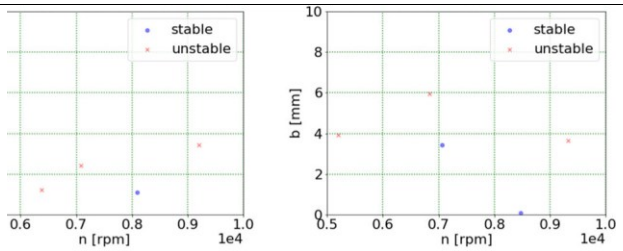
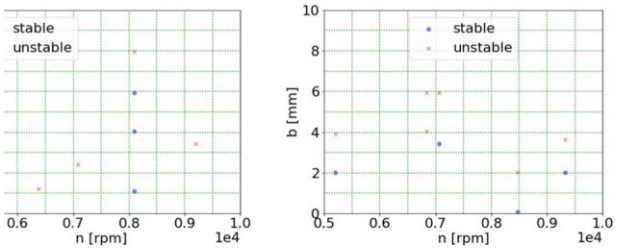
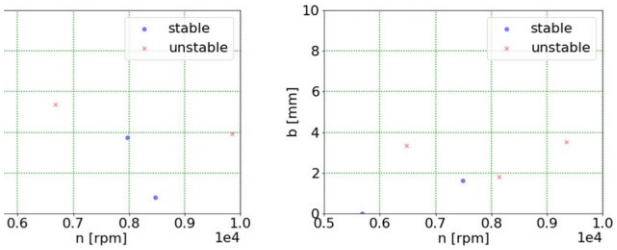


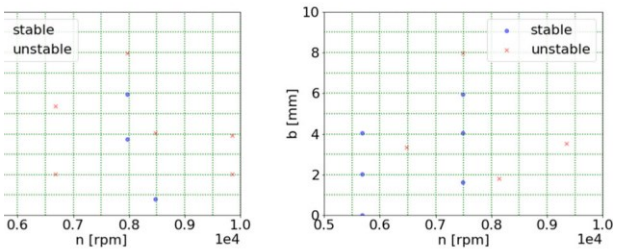
Fig. 30 Sample test points for numerical experiments: Setup 2

2

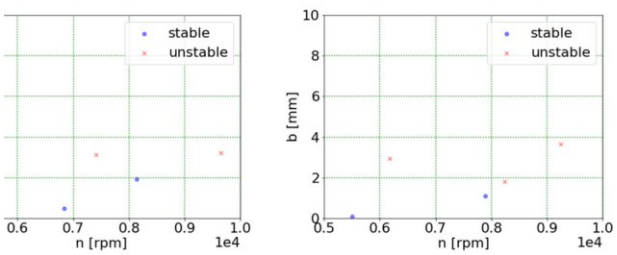




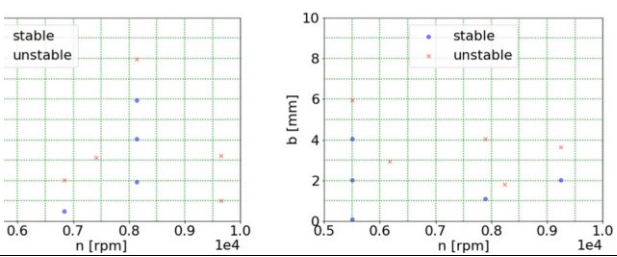




5 tests



10 tests



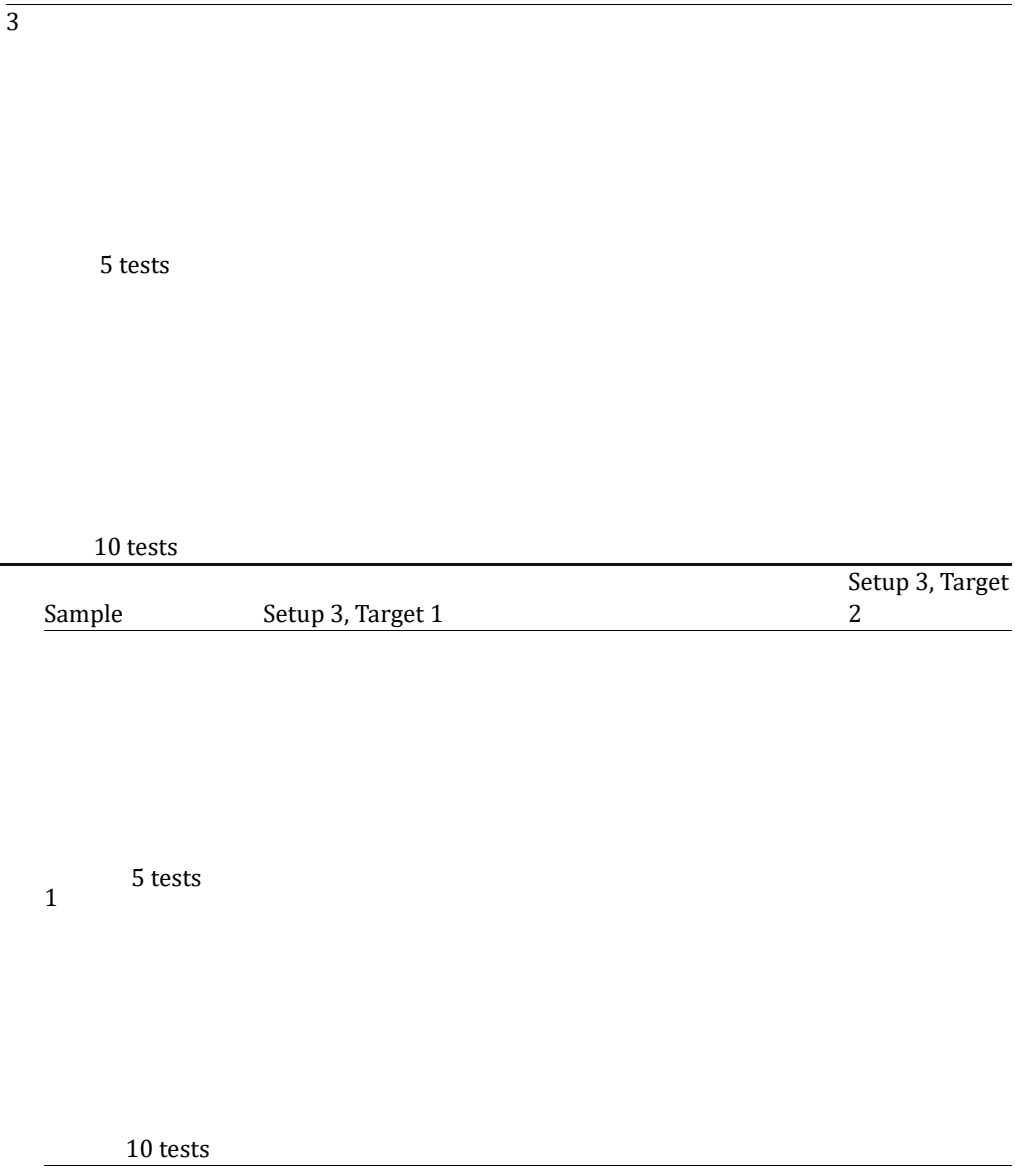
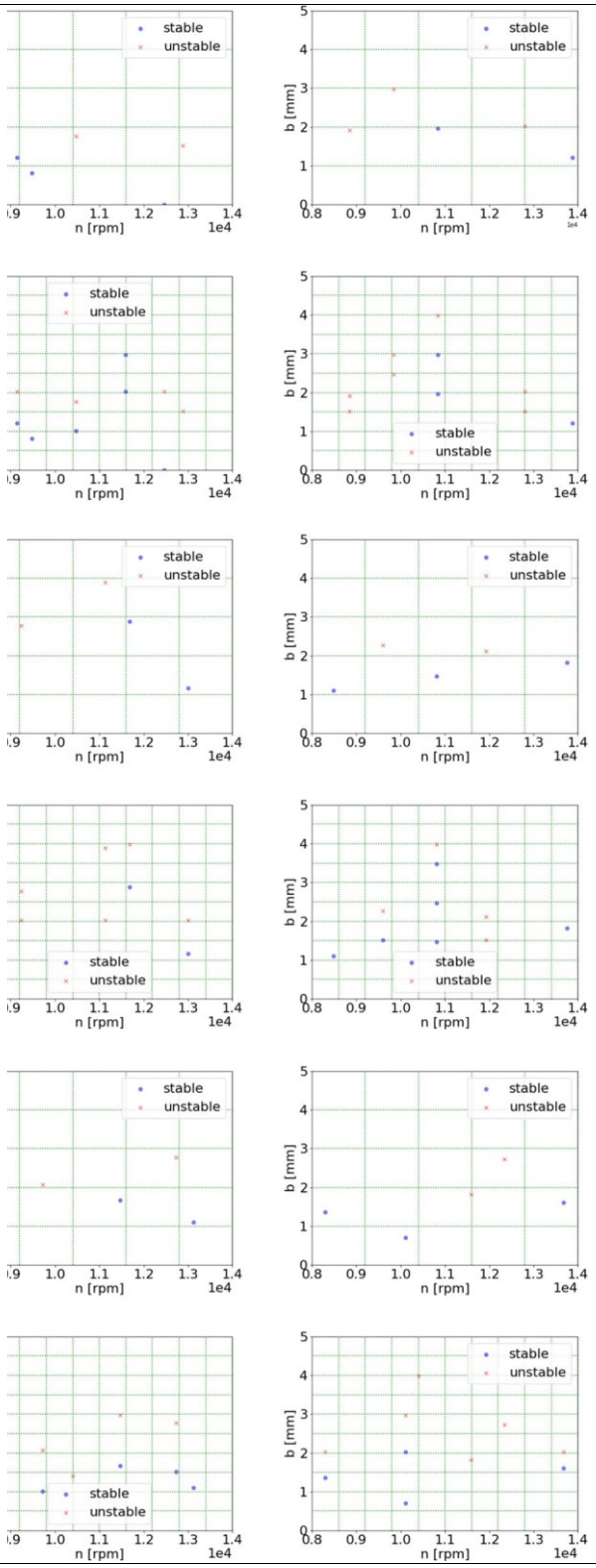


Fig. 31 Sample test points for numerical experiments: Setup 3

2



5 tests

10 tests

References

Akcay, S., Atapour-Abarghouei, A., Breckon, T. P. (2019). Skipganomaly: Skip connected and adversarially trained encoderdecoder anomaly detection. In *2019 international joint conference on neural networks (IJCNN)* (pp. 1–8). IEEE.

Altintas, Y., & Budak, E. (1995). Analytical prediction of stability lobes in milling. *CIRP Annals, 44*(1), 357–362.

Arjovsky, M., Chintala, S., & Bottou, L. (2017). Wasserstein generative adversarial networks. PMLR: In *International conference on machine learning* (pp. 214–223).

Bang, D., & Shim, H. (2018). Improved training of generative adversarial networks using representative features. In *International conference on machine learning* (pp. 433–442).

Bang, D., & Shim, H. (2021). Mggan: Solving mode collapse using manifold-guided training. In *Proceedings of the IEEE/CVF international conference on computer vision* (pp. 2347–2356).

Budak, E., & Altintas, Y. (1998). Analytical prediction of chatter stability in milling-Part I: General formulation.

Campatelli, G., & Scippa, A. (2012). Prediction of milling cutting force coefficients for aluminum 6082–t4. *Procedia Cirp, 1*, 563–568.

Chakraborty, A., Alam, M., Dey, V., Chattopadhyay, A., & Mukhopadhyay, D. (2018). Adversarial attacks and defences: A survey. arXiv preprint arXiv:1810.00069

Chen, G., Li, Y., Liu, X., & Yang, B. (2021). Physics-informed bayesian inference for milling stability analysis. *International Journal of Machine Tools and Manufacture, 167*, 103767.

Cherukuri, H., Perez-Bernabeu, M., Selles, E., & Schmitz, T. (2019). Machining chatter prediction using a data learning model. *Journal of Manufacturing and Materials Processing, 3*(2), 45.

Chong,P.,Ruff,L.,Kloft,M.,&Binder,A.(2020).Simpleandeffective prevention of mode collapse in deep one-class classification. In *2020 international joint conference on neural networks (IJCNN)* (pp. 1–9).

Cornelius, A., Karandikar, J., Gomez, M., & Schmitz, T. (2021). A bayesian framework for milling stability prediction and reverse parameter identification. *Procedia Manufacturing, 53*, 760–772.

Dang, J. W., Zhang, Y., Yang, W. H., & Wan, M. (2010). Cutting force modeling for flat end milling including bottom edge cutting effect. *International Journal of Machine Tools and Manufacture, 50*(11), 986–997.

Deng, C., Tang, J., Lu, S., Ma, Y., Lin, L., & Miao, J. (2023). Improved milling stability analysis for chatter-free machining parameters planning using a multi-fidelity surrogate model and transfer learning with limited experimental data. *International Journal of Production Research*, 1–18.

Deng, C., Tang, J., Miao, J., Zhao, Y., Chen, X., & Lu, S. (2023). Efficient stability prediction of milling process with arbitrary tool-holder combinations based on transfer learning. *Journal of Intelligent Manufacturing, 34*(5), 2263–2279.

Ding, Y., Zhu, L., Zhang, X., & Ding, H. (2010). A full-discretization method for prediction of milling stability. *International Journal of Machine Tools and Manufacture, 50*(5), 502–509.

Espindola, R. P., & Ebecken, N. F. (2005). On extending f-measure and g-mean metrics to multi-class problems. *WIT Transactions on Information and Communication Technologies, 35*, 25–34.

Eynian, M. (2019). In-process identification of modal parameters using dimensionless relationships in milling chatter. *International Journal of Machine Tools and Manufacture, 143*, 49–62.

Friedrich, J., Hinze, C., Renner, A., Verl, A., & Lechler, A. (2017). Estimation of stability lobe diagrams in milling with continuous learning algorithms. *Robotics and Computer-Integrated Manufacturing, 43*, 124–134.

Goodfellow, I. (2016). Nips 2016 tutorial: Generative adversarial networks. arXiv preprint arXiv:1701.00160

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., & Bengio, Y. (2014). Generative adversarial nets. *Advances in Neural Information Processing Systems, 27*.

Greis, N. P., Nogueira, M. L., Bhattacharya, S., & Schmitz, T. (2020). Physics-guided machine learning for self-aware machining. In *2020 AAAI spring symposium on AI and manufacturing*.

Greis, N. P., Nogueira, M. L., Bhattacharya, S., Spooner, C., & Schmitz, T. (2023). Stability modeling for chatter avoidance in self-aware machining: An application of physics-guided machine learning. *Journal of Intelligent Manufacturing, 34*(1), 387–413.

Grossi, N., Sallese, L., Scippa, A., & Campatelli, G. (2015). Speedvaryingcuttingforcecoefficientidentificationinmilling.*Precision Engineering, 42*, 321–334.

Grossi, N., Sallese, L., Scippa, A., & Campatelli, G. (2016). Identification of machine tool dynamics under operational conditions. In *The Proceedings of MTTRF 2016 annual meeting* (pp. 83–88).

Gupta, P., Law, M., & Mukhopadhyay, S. (2020). Evaluating tool point dynamics using output-only modal analysis with mass-change

methods. *CIRP Journal of Manufacturing Science and Technology, 31*, 251–264.
Ham, H., Jun, T. J., & Kim, D. (2020). Unbalanced gans: Pre-training the generator of generative adversarial network using variational autoencoder. arXiv preprint arXiv:2002.02112
Insperger, T., & Stépàn, G. (2002). Semi-discretization method for delayed systems. *International Journal for Numerical Methods in Engineering, 55*(5), 503–518.
Jauhari, K., Rahman, A. Z., Al Huda, M., Widodo, A., & Prahasto, T. (2023). Building digital-twin virtual machining for milling chatter detection based on vmd, synchro-squeeze wavelet, and pre-trained network cnns with vibration signals. *Journal of Intelligent Manufacturing*, 1–32.
Karandikar, J., Honeycutt, A., Schmitz, T., & Smith, S. (2020). Stability boundaryandoptimaloperatingparameteridentificationinmilling using bayesian learning. *Journal of Manufacturing Processes, 56*, 1252–1262.
Lazarou, C. (2020). Autoencoding generative adversarial networks. arXiv preprint arXiv:2004.05472
Munjal, P., Paul, A., & Krishnan, N. C. (2020). Implicit discriminator in variational autoencoder. In *2020 international joint conference on neural networks (IJCNN)* (pp. 1–8).
Oleaga, I., Pardo, C., Zulaika, J. J., & Bustillo, A. (2018). A machinelearningbasedsolutionforchatterpredictioninheavy-dutymilling machines. *Measurement, 128*, 34–44.
Özs¸ahin, O., Budak, E., & Özgüven, H. N. (2015). In-process tool point frf identification under operational conditions using inverse stability solution. *International Journal of Machine Tools and Manufacture, 89*, 64–73.
Özs¸ahin, O., Özgüven, H. N., & Budak, E. (2010). Analysis and compensation of mass loading effect of accelerometers on tool point frf measurements for chatter stability predictions. *International Journal of Machine Tools and Manufacture, 50*(6), 585–589.
Postel, M., Bugdayci, N. B., Monnin, J., Kuster, F., & Wegener, K. (2018). Improved stability predictions in milling through more realistic load conditions. *Procedia CIRP, 77*, 102–105.
Postel, M., Bugdayci, B., & Wegener, K. (2020). Ensemble transfer learning for refining stability predictions in milling using experimental stability states. *The International Journal of Advanced Manufacturing Technology, 107*(9), 4123–4139.
Radford, A., Metz, L., & Chintala, S. (2015). Unsupervised representation learning with deep convolutional generative adversarial networks. arXiv preprint arXiv:1511.06434
Rosca, M., Lakshminarayanan, B., Warde-Farley, D., & Mohamed, S. (2017). Variational approaches for auto-encoding generative adversarial networks. arXiv preprint arXiv:1706.04987
Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., & Chen, X. (2016). Improved techniques for training GANs. *Advances in Neural Information Processing Systems, 29*, 2234–2242.
Schlegl, T., Seeböck, P., Waldstein, S. M., Langs, G., & SchmidtErfurth,U.(2019). f-anogan:Fastunsupervisedanomalydetection with generative adversarial networks. *Medical Image Analysis, 54*, 30–44.
Schmitz, T., Cornelius, A., Karandikar, J., Tyler, C., & Smith, S. (2022). Receptance coupling substructure analysis and chatter frequencyinformed machine learning for milling stability. *CIRP Annals, 71*(1), 321–324.
Schmitz, T., & Smith, S. (2019). *Machining dynamics: Frequency response to improved productivity*. Springer.
Shanavas, N. A., Law, M., & Singh, M. K. (2023). Learning machining stabilitydiagramsfromdatausingneuralnetworks.*Manufacturing Technology Today, 22*(2), 29–41.
Sinwar, D., & Kaushik, R. (2014). Study of euclidean and manhattan distance metrics using simple k-means clustering. *International Journal for Research in Applied Science and Engineering Technology, 2*(5), 270–274.
Totis, G., Insperger, T., Sortino, M., & Stépán, G. (2019). Symmetry breaking in milling dynamics. *International Journal of Machine Tools and Manufacture, 139*, 37–59.
Unver, H. O., & Sener, B. (2023). A novel transfer learning framework for chatter detection using convolutional neural networks. *Journal of Intelligent Manufacturing, 34*(3), 1105–1124.
Wan, M., Feng, J., Ma, Y. C., & Zhang, W. H. (2017). Identification of milling process damping using operational modal analysis. *International Journal of Machine Tools and Manufacture, 122*, 120–131.
Yan, X., Melkote, S., Mishra, A. K., & Rajagopalan, S. (2023). A digital apprentice for chatter detection in machining via human-machine interaction. *Journal of Intelligent Manufacturing, 34*(7), 3039–3052.
Yesilli, M. C., Khasawneh, F. A., & Mann, B. P. (2022). Transfer learning for autonomous chatter detection in machining. *Journal of Manufacturing Processes, 80*, 1–27.
Yin, C., Wang, Y., Ko, J. H., Lee, H. P., & Sun, Y. (2023). Attentiondriventransferlearningframeworkfordynamicmodelguid edtime domain chatter detection. *Journal of Intelligent Manufacturing*, 1– 19.
Zaghbani, I., & Songmene, V. (2009). Estimation of machine-tool dynamic parameters during machining operation through operational modal analysis. *International Journal of Machine Tools and Manufacture, 49*(12–13), 947–957.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this articleis solelygoverned by the terms of such publishing agreement and applicable law.