

Stabilizing Generative Adversarial Networks: A Survey

Maciej Wiatrak

*School of Informatics
University of Edinburgh*

MACWIATRAK@GMAIL.COM

Stefano V. Albrecht

*School of Informatics
University of Edinburgh*

S.ALBRECHT@ED.AC.UK

Andrew Nystrom

Google Research

NYSTROM@GOOGLE.COM

Abstract

Generative Adversarial Networks (GANs) are a type of generative model which have received much attention due to their ability to model complex real-world data. Despite their recent successes, the process of training GANs remains challenging, suffering from instability problems such as non-convergence, vanishing or exploding gradients, and mode collapse. In recent years, a diverse set of approaches have been proposed which focus on stabilizing the GAN training procedure. The purpose of this survey is to provide a comprehensive overview of the GAN training stabilization methods which can be found in the literature. We discuss the advantages and disadvantages of each approach, offer a comparative summary, and conclude with a discussion of open problems.

Keywords: Generative Adversarial Networks, Stabilization, Survey

1. Introduction

Generative Adversarial Networks (GANs) (Goodfellow et al., 2014) are a type of generative model which have received much attention in recent years due to their ability to model complex real-world data. Comprised of two models, a generator (G) and a discriminator (D) both typically represented as neural networks, the two models are trained in a process of adversarial optimization. GANs have been shown to achieve state-of-the-art results in the generation of images (Odena et al., 2017; Radford et al., 2016; Salimans et al., 2016; Zhu et al., 2017), natural language (Yu et al., 2016), time-series synthesis (Donahue et al., 2019; Hyland et al., 2017), and other domains (Creswell et al., 2018; Wang et al., 2017a).

Although GANs achieve state-of-the-art results in many domains, they are known to be notoriously difficult to train, suffering significantly from instability problems (Arjovsky et al., 2017; Arjovsky and Bottou, 2017; Salimans et al., 2016; Goodfellow, 2016). Common failures include mode collapse in which the generator maps different inputs to the same class of outputs which results in highly non-diversified samples; non-convergence due to oscillatory and diverging behaviors during the training of both generator and discriminator; and vanishing or exploding gradients leading to slow learning or a complete stop of the learning process (Goodfellow, 2016).

Recent years have seen a plethora of proposed methods to stabilize GAN training, including modified network architectures, loss functions, convergence-aiding training strategies, and other methods (Wang et al., 2019; Arjovsky et al., 2017; Poole et al., 2016; Salimans

et al., 2016). Despite the significant progress made, to date none of these methods fully solve the instability problem. This instability poses a significant obstacle to the wider adoption of GANs as an off-the-shelf method, because significant manual parameter tuning is typically required to make the method work (Lucic et al., 2018).

The purpose of this article is to provide a comprehensive survey of GAN stabilization methods and to highlight open problems for the progress of the field. We describe the GAN stability problems, identify and categorize existing stabilization methods into five different categories, and include a comparative summary of methods. While each category is presented separately, it should be noted that many of the proposed methods are complementary and can potentially be jointly applied.

After discussing related surveys in Section 1.1, the reader is introduced to the concept of GANs and instability problems in Section 2. Section 3 surveys different GAN variants, outlining strengths and weaknesses of each of the approaches. Section 4 describes widely employed heuristics for GAN stabilization. Finally, Section 5 outlines open problems which provide useful directions for future research.

1.1 Related Surveys

Most surveys devoted to the field of GANs offer a synopsis of the whole field, outlining key concepts, methods and applications of GANs, albeit without an in-detail description of the GAN training procedure and instability problems (Hong et al., 2019; Creswell et al., 2018; Wang et al., 2017a). Lucic et al. (2018) conduct a large-scale empirical study which focused on the performance of a variety of GANs. They demonstrated a need for more consistent and objective evaluation of GANs, as well as for constructing variants less sensitive to hyperparameter tuning. Wang et al. (2019) provide a descriptive, limited overview of the GAN taxonomy, focusing solely on the architecture and loss function variants. Hong et al. (2019) provide an overview of the current GAN landscape, but omit a number of aspects crucial for stabilization of GAN training.

In contrast to the surveys mentioned above, a number of studies focus solely on GAN training issues. This includes the work of Arjovsky and Bottou (2017) who provide a comprehensive outline of the GAN training issues and the underlying theory. Furthermore, a plethora of work deals with practical considerations (Zhang et al., 2018a; Salimans et al., 2016; Huszar, 2016) as well as theoretical aspects (Thanh-Tung et al., 2017; Arora et al., 2018; Fedus et al., 2018; Mescheder et al., 2018; Heusel et al., 2017; Arora et al., 2017; Mescheder et al., 2017) towards training GANs. A concise practical study by Kurach et al. (2018) investigates the impact of a few chosen loss functions, architecture, normalization and regularization schemes. They explore individual and joint performance, as well as sensitivity to hyperparameter tuning.

Our survey provides a comprehensive summary of current GAN techniques and an in-depth study on GAN training problems, providing an overview of the current state of the GAN research landscape with a specific focus on the problem of GAN training stability. It also aims to describe and categorize GAN variants into a holistic taxonomy, and includes an overview of previously overlooked game theory and gradient-based approaches, as well as general training methods.

2. Generative Adversarial Networks

The basic idea of GANs is a competitive game between the generator, G , and discriminator, D . The generator is given a Gaussian or uniform random noise $z \sim p_z$, and transforms it in such a way that $G(z)$ resembles a target distribution p_r . Specifically, the generator is trained to maximize the expected log-probability with which the discriminator classifies $G(z)$ as a real sample, $\mathbb{E}_{z \sim p_z} \log[D(G(z))]$. Conversely, the discriminator is provided with a set of unlabeled samples from $G(z)$ and p_r , and is trained to distinguish between the generated (fake) and real samples. Formally, the discriminator is trained to maximize the expected log-probability $\mathbb{E}_{x \sim p_r} \log[D(x)]$ while simultaneously minimizing $\mathbb{E}_{z \sim p_z} \log[D(G(z))]$. This competition between G and D can be formulated as a two-player zero-sum game in which the following loss function is maximized by the discriminator and minimized by the generator:

$$L(D, G) = \mathbb{E}_{x \sim p_r} \log[D(x)] + \mathbb{E}_{z \sim p_z} \log[1 - D(G(x))] \quad (1)$$

In their original publication, Goodfellow et al. (2014) proved the existence of a unique solution at $D = \frac{1}{2}$, which since GANs are a zero-sum two-player game, corresponds to a Nash equilibrium (NE) in which neither player can improve their cost unilaterally (Nash, 1951). In practice, however, GANs have been shown to struggle to reach this NE (Fedus et al., 2018; Salimans et al., 2016). Furthermore, the evaluation and benchmarking of GANs has been hampered by the lack of a single universal and comprehensive performance measure.

2.1 Instability Problems in GAN Training

Despite the proof of existence of the unique equilibrium in the GAN game, the training dynamics of GANs have proven to be notoriously unstable (Arjovsky et al., 2017). We now discuss the main sources of instability.

In their original formulation, Goodfellow et al. (2014) have shown that GANs optimize the Jensen-Shannon divergence (JSD)

$$\text{JSD}(\mathbb{P}_r || \mathbb{P}_g) = \frac{1}{2} \text{KL}(\mathbb{P}_r || \mathbb{P}_A) + \frac{1}{2} \text{KL}(\mathbb{P}_g || \mathbb{P}_A) \quad (2)$$

where KL denotes the Kullback-Leibler divergence and \mathbb{P}_A is defined as the ‘‘average’’ distribution $\mathbb{P}_A = \frac{\mathbb{P}_r + \mathbb{P}_g}{2}$. Using the definition of JSD and \mathbb{P}_A , the optimal discriminator D^* is

$$D^* = \frac{P_r(x)}{P_r(x) + P_g(x)}. \quad (3)$$

Incorporating D^* into the discriminator’s loss as a function of θ , it can be shown that at optimality, the discriminator is minimizing the following equation

$$L(D^*, g_\theta) = 2\text{JSD}(\mathbb{P}_r || \mathbb{P}_g) - 2 \log 2. \quad (4)$$

The concept of an optimal discriminator is particularly important because its optimality would ensure that the discriminator produces meaningful feedback to the generator. **Therefore to achieve this, it is reasonable to assume that a training procedure would include training the discriminator to optimality for each iteration of the generator.** Nevertheless, extensive practical (Salimans et al., 2016) and theoretical (Arjovsky et al., 2017) studies have shown that training the discriminator to optimality works only in theory. The following sub-sections detail the major causes of this instability.

2.1.1 CONVERGENCE

Although the existence of a global NE has been proven (Goodfellow et al., 2014), arriving at this equilibrium is not straightforward. The proof of convergence described above does not hold in practice due to the fact that the generator and discriminator are modeled as neural networks; thus, the optimization procedure operates in the parameter space of these networks, instead of directly learning the probability density function. Furthermore, the non-convex-concave character of the game makes it especially hard for the gradient descent ascent (GDA) algorithm to converge, often resulting in diverging, oscillating, or cyclic behavior (Mertikopoulos et al., 2017; Salimans et al., 2016; Goodfellow, 2015), and proneness to convergence to a local NE within the parameter space of the neural networks. Furthermore, the local NE can be arbitrarily far from the global NE, thus preventing the discriminator from being optimal. Finally, from a practical standpoint, training the discriminator to optimality for every generator iteration is highly computationally expensive.

2.1.2 VANISHING OR EXPLODING GRADIENTS

According to the theory laid out at the beginning of this section, a discriminator close to optimality would produce meaningful feedback to the generator, thus resulting in an improved generator. However, in practice the cost of the well trained discriminator is not $2\text{JSD}(\mathbb{P}_r||\mathbb{P}_g) - 2\log 2$, but rather approaches 0. Such a highly accurate discriminator, where $D(x) = 1, \forall x \in p_r$ and $D(G(z)) = 0, \forall G(z) \in p_z$ squashes the loss function to 0, producing gradients close to zero, which provides little feedback to the generator, slowing or completely stopping the learning. Hence, as the discriminator gets better at discerning between real and generated samples, the updates to the generator get consistently worse. One of the reasons behind the discriminator easily overpowering the generator is that the supports of real and generated distributions often lie on low dimensional manifolds, making it possible for a perfect discriminator to exist. In such a scenario of a perfect discriminator, its gradients will approach zero on almost every data point, hence providing no feedback to the generator. To avoid the vanishing gradients problem, Goodfellow et al. (2014) proposed to use an alternative loss for the generator, namely $-\log D(G(z))$, which is widely used in practice and referred to as non-saturating (NS) loss. Although this loss modification alleviates the vanishing gradients problem, change to the loss has not completely solved the problem, in turn contributing to more unstable and oscillating training.

2.1.3 MODE COLLAPSE

One of the most common failures in GANs, *mode collapse* happens when the generator maps multiple distinct inputs to the same output, which means that the generator produces samples of low diversity (Huszar, 2016; Theis et al., 2015). This is a problem for GAN training, since a generator without the ability to produce diverse samples is not useful for training purposes. Discovering and tackling mode collapse is a difficult problem, since the cause of mode collapse is rooted deeply in the concept of GANs. Metz et al. (2017) argued that with the generator greedily optimizing its loss function in order to “fool” the discriminator, each generator update is a partial collapse towards mode collapse. An intuitive solution to mode collapse is a well trained discriminator capable of robustly identifying the symptoms of mode collapse. However, as discussed above, this runs counter to the vanishing or ex-

ploding gradients problem. A crucial aspect in addressing mode collapse is a discriminator with high generalization capabilities (Arora et al., 2017; Thanh-Tung et al., 2017)

All of the above problems contribute to the instability of the GAN training process. These problems are often highly linked with each other and addressing them often takes character of a trade-off. A crucial challenge is the accurate estimation of the generalization capabilities of the discriminator, which in turn would make it possible to properly evaluate the generator. A final obstacle is lack of a comprehensive evaluation metric. At the current state, a number of evaluation metrics have been proposed, such as *Inception* or *Frechet Inception Distance* (FID). Nevertheless, none of them offer a unique solution (Grnarova et al., 2018; Lucic et al., 2018).

3. GAN Variants and Stabilization Techniques

To categorize different methods focused on improving GAN training, the following sections divide the literature into five approaches (cf. Figure 1): architecture, loss function, game theory, multi-agent, and gradient-based. We summarize the key concepts of each approach and discuss their overall advantages and disadvantages.

3.1 Modified Architectures

The most extensively studied category is architecture variant GANs. This category covers methods in which the proposed improvements involve networks or training structure while keeping the loss function largely intact from the previously proposed methods. The majority of the architecture-based variants have been established for a specific application (Creswell et al., 2018; Hitawala, 2018; Zhu et al., 2017; Yu et al., 2016). Here, we focus on the more general approach, which has shown to improve stability of the training procedure in GANs.

3.1.1 CONVOLUTIONAL ARCHITECTURES

One of the most popular and successful architecture-variant GANs are inspired by computer vision techniques, specifically Convolutional Neural Networks (CNNs) (Krizhevsky et al., 2012). An example which has been shown to significantly improve the quality of generated images is the Deep Convolutional GAN (DCGAN) (Radford et al., 2016). DCGAN applies a series of convolutional operations including a spatial up-sampling operation to improve the generator. Having shown to enhance the quality of the produced samples solely through architectural changes, DCGAN is often treated as a baseline for modeling other GAN variants (Hong et al., 2019). A further improvement was proposed by Zhang et al. (2019) in Self-Attention GAN (SAGAN), who incorporated a self-attention mechanism into the design of both networks in order to capture local and global dependencies of the target distribution. Finally, BigGAN (Brock et al., 2018) builds on SAGAN and introduces a number of incremental improvements. In particular, both the batch size and number of model parameters are increased, contributing to BigGAN obtaining state-of-the-art results on the ImageNet.

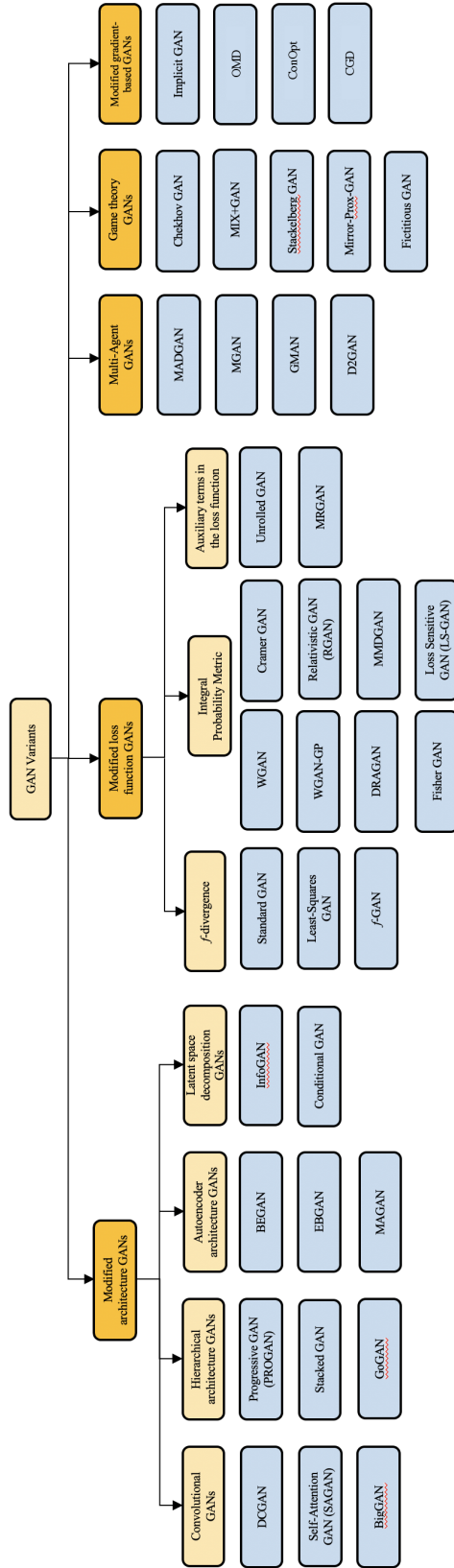


Figure 1: Taxonomy of GAN methods.

3.1.2 HIERARCHICAL ARCHITECTURES

A difficult problem in GAN training is the discriminator overpowering the generator in the early stages of the game, thus failing to properly train the generator. This is particularly challenging for generating high resolution images. A heuristic approach designed to tackle this problem is to allow the generator to make a number of updates per one discriminator update (Goodfellow, 2016). Nevertheless, this technique has been shown to be limited and often leading to other instability problems. Progressive GAN (PROGAN) (Karras et al., 2018) addresses this problem by training the networks in multiple phases, starting from a low-resolution image and minimal network architecture, progressively expanding both networks. Additionally, PROGAN proposes a number of training techniques improving stability for image generation, including pixel normalization and mini-batch standard deviation.

Another hierarchical approach is the simultaneous training of multiple GAN pairs. Stacked GAN (SGAN) (Huang et al., 2017) stacks pairs of generator and discriminator, each consisting of an encoder and decoder on top of each pair. Apart from calculating standard adversarial loss, at each stack the encoder and decoder calculate the discrepancy between generated and real image (referred to as conditional loss) as well as entropy loss, which prevents mode collapse. All three losses are then combined into a composite loss and used in the joint training of all stacks. Similar to Stacked GAN, GoGAN (Juefei-Xu et al., 2017) trains pairs of WGANs (Arjovsky et al., 2017) (see Section 3.2.2), but uses margin based Wasserstein distance, instead of standard Wasserstein distance. This makes it possible for the discriminator to focus on samples with smaller margin and progressively reduce the margin at each stage of the game. Therefore, steadily decreasing the discrepancy between the generated and target distribution.

3.1.3 AUTOENCODER ARCHITECTURES

Another approach designed to prevent the discriminator from optimizing greedily (which may overpower the generator at early stages of the training) is the adoption of an autoencoder architecture for the discriminator. Instead of outputting a probability to distinguish between generated (fake) and real samples, the discriminator assigns an energy value to given samples. The energy model is attained by feeding a sample (generated or real) to a discriminator consisting of an encoder and a decoder, calculating its reconstruction cost, and training it to output high energy level for generated and low for real samples. Intuitively, the autoencoder architecture takes into account the complexity of the image. Furthermore, as the autoencoder is trained to focus on reconstructing the most important features (i.e. near the data manifold), it provides feedback on these features for the generator. The autoencoder architecture was initially proposed by Zhao et al. (2017), with the loss function being the energy level. Additionally, in order to stabilize training the discriminator ignored the sample outliers with high reconstruction error. The margin restriction was later relaxed by Wang et al. (2017b) in MAGAN, which argued that a fixed margin has a negative influence on the dynamics of the GAN. More complex Boundary Equilibrium GAN (BEGAN) (Berthelot et al., 2017) uses Wasserstein distance to match the distributions between reconstructed and generated or real examples. Moreover, it adds a hyperparameter $\gamma = \frac{\mathbb{E}[L(G(z))]}{\mathbb{E}[L(x)]}$ to balance the training between the generator and the discriminator.

3.1.4 LATENT SPACE DECOMPOSITION ARCHITECTURES

One of the reasons behind the instability of training is the low-dimensional support of generated and target distributions, which can lead to the greedy optimization of the discriminator (Arjovsky and Bottou, 2017). While the dimensions of p_r appear to be high, in reality they are usually concentrated in lower dimensional manifolds, which often mirror the most important features of the distribution (for example, a feature of a human face on an image). Such lower dimensional manifolds establish a large number of restrictions for the generator, often making it unattainable for the generator to follow them. This contributes to the disjointedness of generated and real distribution, thus making it easier for the discriminator to perfectly discriminate between the samples.

Latent space decomposition methods aim at providing additional information to the input latent vector z , enabling it to disentangle the relevant features from the latent space. InfoGAN (Chen et al., 2016) adopts an unsupervised method which supplies a latent variable c capturing the meaningful features of the real sample on top of the standard input noise vector z . InfoGAN then maximizes the mutual information $I(c, G(z, c))$ between z and c , transforming the generated sample with $G(z, c)$. Due to intractability of estimating the posterior $p(c|x)$, the algorithm adopts a variational approach maximizing a lower bound of $I(c, G(z, c))$, instead of directly calculating it.

In contrast to InfoGAN, Conditional GAN (CGAN) (Odena et al., 2017) proposes a supervised method, adding a class label c to a generator and discriminator. This approach has been shown to produce high quality images and prevent mode collapse. However, this is with the strong restriction of possessing a labeled dataset.

3.1.5 SUMMARY

All of the architecture variant GANs and techniques described above have been shown to improve the stability of training in GANs. Nevertheless, the scale at which each of the methods manages to improve the training differs widely. A current strong baseline is BigGAN, which enhances the training in all of the aspects mentioned in the previous section. This improvement, however, is attained at a great computational cost, and improving the training procedure without an increase in complexity remains a challenge. Architecture-variant GANs is a developing and dynamic field, often offering low-hanging-fruits in terms of performance improvements, especially for application-specific GANs. Nevertheless, the non-convex-concave character of the GAN game is a challenge that will require improvements beyond architectural changes.

3.2 Modified Loss Functions

The shortcomings of the standard GAN loss function were already identified by Goodfellow et al. (Goodfellow, 2015; Goodfellow et al., 2014), who showed that KL divergence and JS divergence under idealized conditions (Fedus et al., 2018) contributes to oscillating and cyclical behavior, especially when the distributions of generated and target data are disjoint. This leads to the non-convergence and vanishing gradients problems. A large number of studies have been devoted to this problem (Wang et al., 2019; Lucic et al., 2018; Barnett, 2018; Metz et al., 2017; Arjovsky and Bottou, 2017; Poole et al., 2016). In this section, we

introduce a number of alternative loss functions employed for GANs and highlight the most important ones.

3.2.1 f -DIVERGENCE

The KL and JSD divergences discussed in the standard GAN (Goodfellow et al., 2014) belong to an f -divergence family, which measures the difference between two given probability distributions. Given distributions p_r and p_g , the f -divergence with a function f is defined as:

$$D_f(p_r \parallel p_g) = \int_{\mathcal{X}} p_g(x) f\left(\frac{p_r(x)}{p_g(x)}\right) dx \quad (5)$$

where the generator function f is a convex function and satisfies the condition $f(1) = 0$. In other words, the function f becomes 0 when two distributions are equivalent. The f -divergence family can be indirectly calculated by estimating the expectation of the lower bound, which circumvents the intractability problem of unknown probability distributions. Nowozin et al. (2016) discussed the GAN loss function in terms of various divergences and their efficacy under an arbitrary function f , proposing f-GAN, which generalizes the GAN loss function through the estimation of various f -divergences given f . Another f -divergence based GAN is the Least Squares GAN (LSGAN), which tries to remedy the problem of vanishing gradient by replacing the sigmoid cross-entropy loss present in the standard GAN with the least-squares loss, penalizing generated samples far from the decision boundary (Mao et al., 2016). As a divergence, LSGAN has been proven to use the Pearson χ^2 divergence.

Nevertheless, the f -divergence family is subject to a number of limitations. In particular, the growing dimension of data makes it harder to estimate the divergence and the support of the two distributions becomes low dimensional, resulting in the divergence value going to infinity (Sriperumbudur et al., 2009).

3.2.2 INTEGRAL PROBABILITY METRIC

Designed to address the restrictions of the f -divergence family, integral probability metrics (IPM) methods are not subject to the data dimension and distribution disjointedness problems, resulting in a consistent distance between the data distributions. Furthermore, in contrast to f -divergence in which the discriminator is a binary classifier, the discriminator in an IPM framework is a regression task providing a scalar value and henceforth opening a new avenue for research by treating GANs as an actor-critic problem (Pfau and Vinyals, 2017). Defined as a critic function f in a function class \mathcal{F} consisting of real-valued, bounded, measurable functions, an IPM measures the maximal distance between two distributions. Given p_r and p_g on a compact space \mathcal{X} , the IPM metric can be denoted as:

$$d_{\mathcal{F}}(p_r, p_g) = \sup_{f \in \mathcal{F}} \mathbb{E}_{x \sim p_r}[f(x)] - \mathbb{E}_{x \sim p_g}[f(x)] \quad (6)$$

One of the distances which belong to IPM is the *Wasserstein* or Earth Mover's (EM) distance, which is used in Wasserstein GAN (WGAN) (Arjovsky et al., 2017) as the new loss measure. Wasserstein distance is defined as:

$$W(p_r, p_g) = \inf_{\gamma \in \Pi(p_r, p_g)} \mathbb{E}_{(x,y) \sim \gamma} [\|x - y\|] \quad (7)$$

Wasserstein distance has been shown to significantly improve training stability and convergence, dealing particularly well with the distributions support lying on low dimensional manifolds. However, due to the intractability of the infimum term, the critic f needs to be parameterized and weight clipping needs to be applied. A significant drawback is that weight clipping incurs pathological behavior that contributes to slow convergence and not entirely stable training. These problems have been partly addressed by including a gradient penalty instead of weight clipping in the loss function of WGAN-GP (Gulrajani et al., 2017). Another shortcoming of the WGAN was discussed by Bellemare et al. (2017) who argued that the WGAN incurs biased gradients. Their solution to this problem suggests an energy function similar to Wasserstein distance, but without the biased gradients. Furthermore, the Cramer distance is also linked with the kernel embedded space distance, discussed in the next paragraph on Maximum Mean Discrepancy.

Derived from Generative moment matching networks (GMMN) (Li et al., 2015), Maximum Mean Discrepancy (MMD) distance is an alternative to Wasserstein distance. Continuous and differentiable like the Wasserstein distance, MMD distance has been applied in Maximum Mean Discrepancy GAN (MMDGAN), which can be seen as a combination of GMMNs and GANs (Li et al., 2017). Essentially, MMD measures the distance between the means of the embedding space of two distributions, using the kernel trick with a Gaussian kernel. In MMDGAN, the Gaussian kernels are replaced by adversarial kernel learning techniques, which is claimed to better represent the feature space. This can be seen as advantageous over WGAN. However, the computational complexity of MMDGAN which increases exponentially with the number of samples is a significant drawback.

A different IPM approach is adopted in Relativistic GAN (RGAN) (Jolicœur-Martineau, 2018; Jolicœur-Martineau, 2019), in which the authors argue for a general approach to devising new GAN loss functions. This approach estimates the probability that a generated sample is real, taking into account the probability of a corresponding real sample being real. In other words, we measure the distance between the probability of generated sample being real and the probability of real sample being real. It has been shown that the RGAN approach contributes to overcoming the greedy optimization of the discriminator. Furthermore, the technique outlined in RGAN could be potentially expanded to other GANs.

Other IPM approaches include Geometric GAN (Lim and Ye, 2017) which draw on the Support Vector Machine (SVM) algorithm (Cortes and Vapnik, 1995), utilizing separating hyperplanes to update the generator and discriminator and encouraging the generator to move toward the separating hyperplane and the discriminator to move away from it. Although proven to improve training stability, the Geometric GAN requires expensive matrix computation. Loss Sensitive GAN (LS-GAN) (Qi, 2018) is based on the idea that the loss for a real sample should be smaller than the loss for a generated sample, and aims at minimizing the margins between the two. The reasoning behind LS-GAN is that the non-parametric assumption of the discriminator having infinite capacity (Goodfellow et al., 2014) leads to instability problems, including vanishing gradients, and argues for a more general margin-based approach. Finally, a more stable and computationally efficient approach has been proposed in Fisher GAN (Mroueh and Sercu, 2017), which incorporates a data-dependent constraint into the loss function. Fisher GAN aims at not only reducing the distance between two distributions, but also the in-class variance of them. Furthermore, it avoids adding weight clipping or gradient penalty.

3.2.3 AUXILIARY LOSS FUNCTIONS

The modifications in loss function are not limited to devising new divergences from the f -divergence or IPM family. An important avenue is providing supplementary terms to the already existing GAN loss function in order to improve the training procedure. Here, we highlight two such methods. The first one is inspired by the recursive reasoning opponent modeling methods (Metz et al., 2017), in which the generators “unrolls” possible discriminator reactions and accounts for them by adding a term to the gradient. Given the parameters of the discriminator θ_D and generator θ_G , the surrogate loss by unrolling N steps can be defined as:

$$f_N(\theta_D, \theta_G) = f(\theta_D^N(\theta_D, \theta_G), \theta_G) \quad (8)$$

The intuition is that capturing how a discriminator would react given a change in discriminator and feeding it into a generator would allow the generator to react, and prevent mode collapse. A different approach towards preventing mode collapse is offered in Mode Regularized GAN (MRGAN), which uses an encoder E to produce the latent vector z from a real sample x , $E(x) : x \rightarrow z$ instead of random input noise and alters the loss function by including a term ensuring that different modes are captured in the latent vector z .

3.2.4 SUMMARY

Loss function variant GANs are the most heavily studied type of GANs after the architecture type to date. Some of the loss function variant GANs have managed to predominantly overcome the vanishing gradient problem (Arjovsky et al., 2017; Jolicoeur-martineau, 2019; Miyato et al., 2018; Mao et al., 2016; Qi, 2018). Furthermore, in comparison to architecture types, they offer more generalization properties, with methods such as WGAN or SN-GAN being widely applied and offering avenues for further research (Pfau and Vinyals, 2017). Additionally, the trade-off between the quality of generated data and computational complexity does not show to be as proportional in comparison to some of the architecture variant GANs such as BigGAN (Wang et al., 2019; Brock et al., 2018). However, loss function variant GANs often work only under specific restrictions (Arjovsky et al., 2017; Gulrajani et al., 2017), and proper hyperparameter tuning of architectural design approaches often appear to yield superior results (Lucic et al., 2018). Hence, the stabilization of the training procedure remains a challenge.

3.3 Game Theory for GANs

At the core of all GANs is a competition between generator and discriminator. Game theory GANs focus specifically on that aspect, drawing on the rich literature in the field of two-player competitive games (Freund and Schapire, 1999; von Neumann and Morgenstern, 2007) to aid convergence.

A leading theme in this research area is the computation of **mixed strategy Nash equilibria (MNE)** (Hsieh et al., 2018; Ge et al., 2018; Oliehoek et al., 2018, 2017; Grnarova et al., 2017; Arora et al., 2017). In contrast to the pure strategy Nash equilibrium used in standard GAN, which has been proven to guarantee convergence only to a local Nash equilibrium (Unterthiner et al., 2018; Heusel et al., 2017), **under certain circumstances an MNE is always equal to a global NE**. A proof for the existence of the MNE in the GAN game

together with a sketch of the suitable algorithm was given by Arora et al. (2017). Drawing on this work, Grnarova et al. (2017) proposed a convergent algorithm called ChekhovGAN for finding MNE using a regret minimization algorithm (Hazan et al., 2017; Freund and Schapire, 1999). However, the ChekhovGAN is only provably convergent under the heavy restrictions that the discriminator is a single-layered network. The authors of (Oliehoek et al., 2018, 2017) formulated the game as a finite zero-sum game with a finite number of parameters and proposed using a Parallel Nash Memory algorithm (Oliehoek et al., 2006) to arrive at the MNE. Their work was further extended by a less restrictive use of Mirror Descent in computing MNEs (Hsieh et al., 2018). A relatively simpler yet effective method to compute MNEs was proposed by (Ge et al., 2018), which trains the model using a mixture of historical models. Some of the game theory inspired methods are also present in other approaches (Kodali et al., 2017; Salimans et al., 2016).

3.3.1 SUMMARY

At the current stage, the game theory variant GANs literature is limited, with its proposed methods highly restrictive, thus rarely directly applicable. Nevertheless, work such as (Arora et al., 2017) provides excellent theoretical analysis for tackling problems in the training procedure and opens an avenue for further research.

3.4 Multi-Agent GANs

Multi-agent type GANs extend the idea of using a single pair of generator and discriminator to the multi-agent setting and show how employing such methods can provide better convergence properties and prevent mode collapse. The majority of the literature focuses on introducing a larger number of generators (Zhang et al., 2018b; Arora et al., 2017; Ghosh et al., 2017; Hoang et al., 2017) or discriminators (Albuquerque et al., 2019; Nguyen et al., 2017; Durugkar et al., 2017).

MAD-GAN (Ghosh et al., 2017) uses multiple generators and one discriminator not only to identify fake images but also to determine the specific generator which produced the images fed to it. Aimed at preventing mode collapse, MAD-GAN modifies the generator’s objective function to encourage different generators to produce different samples. It also changes the discriminator to account for these generators and provide feedback to them accordingly to various modes. An extension of MAD-GAN is MGAN (Hoang et al., 2017) which uses a separate classifier in parallel with discriminator to match the generated output with an appropriate generator. Also providing some theoretical justification for the MGAN enforcing the generators to produce diverse samples, thus prevent mode collapse. Stackelberg GAN (Zhang et al., 2018b), drawing on the concept of *Stackelberg competition* (Sinha et al., 2018), offers an architecture design where multiple generators act as followers to the discriminator (leader), utilizing sampling schemes beyond the mixture model.

At the other side of the spectrum, Durugkar et al. (2017) use many discriminators to boost the learning of the generator, while Nguyen et al. (2017) train exactly two discriminators which separately compute KL and reverse KL divergence to place a fair distribution across the data modes.

MIX+GAN (Arora et al., 2017) uses a mixture of multiple generators and discriminators with multiple different parameters to compute mixed strategies. The total reward is

then calculated by taking a weighted average of rewards over the pairs of generators and discriminators. An interesting approach is offered by Ghosh et al. (2016) who examine how multi-agent communication can regularize the training of both networks. Previously discussed SGAN (Huang et al., 2017) and GoGAN (Juefei-Xu et al., 2017) train GANs in local pairs to establish a global GAN pair trained against all of these pairs at the same time. Similarly, Im Jiwoong et al. (2016) also adopts the concept of multiple GAN pairs, but with discriminators dynamically exchanged during training, thus reducing the effect of coupling which can lead to mode collapse.

3.4.1 SUMMARY

The current literature on multi-agent type GANs mainly revolves around the relatively simple idea of hindering mode collapse through the introduction of a variety of generators and discriminators. Although the results are promising, often the computational costs pose a significant problem. Moreover, it is not yet entirely clear how an appropriate number of generators or discriminators should be chosen. Finally, there remain a variety of methods in multi-agent learning literature which have not been exhausted in terms of stabilizing GAN training (Papoudakis et al., 2019; Lowe et al., 2017; He et al., 2016).

3.5 Modified Gradient Optimization

An interesting research avenue is the development of alternatives to the regular gradient descent ascent (GDA) that are more suitable for general games with multiple losses interacting with each other. Regular GANs trained with GDA have demonstrated diverging and oscillating behaviors during training (Mertikopoulos et al., 2017), and although its convergence has been proven using a two-time scale update rule (Heusel et al., 2017), this requires mild assumptions and can only be guaranteed to arrive at a local Nash equilibrium. Daskalakis et al. (2017) used Optimistic Mirror Descent (OMD) to approximately converge, but only under a restriction that the game is a two-player bilinear zero-sum game. Mescheder et al. (2018) proposed the *ConOpt* algorithm that adds a penalty for non-convergence. Based on this result, Balduzzi et al. (2018) and later Letcher et al. (2019) decomposed the game into two components: a *potential game* which can be solved by GDA, and a *Hamiltonian game* suitable for solving with *ConOpt* (Mescheder et al., 2018). Finally, Schäfer and Anandkumar (2019) combined the approaches above and drew on multi-agent methods (Papoudakis et al., 2019) to create a *Competitive Gradient Descent*, suitable specifically for two-player competitive games. This algorithm was later applied to competitive regularization in GANs (Schäfer et al., 2019), where the generator and discriminator act as agents under local information showing more consistent and stable performance.

3.5.1 SUMMARY

The initial results for GANs with gradient-based variants show improvements in convergence and preventing mode collapse in comparison to GANs employing GDA. Although these results could possibly be achieved with suitable hyperparameter tuning, stabilization of the training is a reasonable contribution. Gradient-based methods remain a largely unexplored field with promising research directions. With GDA’s shortcomings in general

games, gradient variants provide an additional edge, and importantly, could be used in line with other methods.

3.6 Other Modifications

This section mentions other GAN methods which have no clear correspondence to the previous categories. One is the Evolutionary GAN (Wang et al.), which adapts the GAN to the environment through a series of operations on the population of generators and discriminators. AdaGAN (Tolstikhin et al., 2017) uses a **boosting technique to iteratively address the problem of missing modes**. An interesting and deeply theoretical approach was proposed by Unterthiner et al. (2018) who provided a proof and a method to escape local equilibria using energy-based potential fields. Another technique for preventing mode collapse includes **marginalizing the weights of the generator and discriminator offered in the Bayesian GAN** (Saatchi and Wilson, 2017).

3.7 Comparative Summary

To conclude the review of the GAN variants, we outline the advantages and disadvantages as well as the outlook of each of the types. Currently, the approach showing the highest empirical improvement is the architecture type. Nevertheless, this is also due to a large number of publications in the area of modified network architecture relative to other GAN variants, low entry barrier, and some of the designs resulting in meticulous parameter optimization. Furthermore, much of the research of this type lacks rigorous theoretical justification and struggles to alleviate all of the training problems, usually showing improvement in only one or two of them (Wang et al., 2019). At the same time, loss function variant GANs, although often well justified, such as WGAN (Arjovsky et al., 2017), show that their results could also be attained by appropriate hyperparameter optimization of architecture variant GANs such as DCGAN. A comprehensive comparison of GANs (Lucic et al., 2018) has demonstrated a small improvement in general metrics by state-of-the-art loss type GANs in comparison to architectures such as DCGAN (Radford et al., 2016), and that it can be more worthwhile to focus on tuning hyperparameters rather than introducing novel methods.

Another survey on convergence in GANs (Fedus et al., 2018) highlighted that much of the rigorous theoretical analysis is often not fulfilled in reality; GANs can converge in situations they theoretically should not, and vice versa. It is hypothesized that this is mostly due to the nature of gradient descent ascent. Loss function variant GANs hold a big promise, addressing an obvious issue which is the unsuitability of the standard objective function to the GAN game. Furthermore, they often demonstrate more general improvement in training than architecture type GANs. We believe, however, that a significant breakthrough in loss function GANs would require incorporating an alternative to GDA. This could possibly be addressed by the gradient-based variants, which might have the broadest influence, enhancing all types of GANs by introducing a gradient design relevant to the GAN game. Nevertheless, gradient-based methods are still at a too early stage to form any reliable predictions. Similarly to loss function variants, game theory variants could provide an approach capable of the general improvement in the training procedure, addressing the problem of dealing with the non-convex-concave character of the game. However, current methods in this field work only under strict limitations and cannot compete with architecture and loss

function type GANs. It should be noted that this area is also in early stages, with only a handful of studies published. Finally, multi-agent GANs have mostly focused on preventing mode collapse, showing promising but not state-of-the-art results in this problem. A vast area of multi-agent learning including actor-critic methods and opponent modeling Albrecht and Stone (2018) remains unexplored.

4. General Stabilization Heuristics

A large scale study on GANs (Lucic et al., 2018) has shown that optimizing hyperparameters tends to yield superior results compared to applying specific methods such as modified architectures and loss functions. Furthermore, other studies in this area (Fedus et al., 2018; Kurach et al., 2018) have demonstrated the impact of regularization and normalization techniques on various loss function and architecture GAN variants. In this section, we discuss general heuristics which have demonstrated consistent stabilization improvements in GAN training. These heuristics can potentially be applied to different GAN types.

4.1 Regularization

The most prominent regularization technique in GANs is the gradient penalty, which has been initially applied to WGAN in WGAN-GP (Gulrajani et al., 2017) as a soft penalty constraining the objective function to 1-Lipschitzness. In this study the authors used linear interpolation between real and generated distributions to calculate the gradient norm used for the penalty. The gradient penalty in WGAN was later extended by Kodali et al. (2017), who looked at the gradient penalty from the perspective of regret minimization. Therefore, using a no-regret type algorithm to constrain the gradient penalty to be calculated only around a data manifold, thus limiting the discriminator functions to the set of linear functions. A different approach was adopted by Roth et al. (2017) who proposed a gradient penalty on the weighted gradient-norm of the discriminator, designed specifically to tackle the issue of non-overlapping support between the generated and target distribution in GANs from f -divergence family.

4.2 Normalization

Normalization of the discriminator can stabilize the GAN training procedure from two perspectives. First, it can lead to more stable optimization through inducing the discriminator to produce better quality feedback. Second, it can enrich the representation within the layers of the network of the discriminator.

Optimization-wise, two techniques have been particularly applied to the GAN stability problem. These are batch normalization (Ioffe and Szegedy, 2016) and layer normalization (Ba et al., 2016). Batch normalization, introduced by Denton et al. (2015) and popularized by Radford et al. (2016), has been shown to greatly improve the optimization of neural networks through normalizing the activations of each layer, allowing each layer to learn more stable distribution of inputs. Batch normalization was further extended by Salimans et al. (2016) who proposed virtual batch normalization, which aims to make the input sample less dependent on other samples in the batch, thereby alleviating the mode collapse problem. In contrast to batch normalization, the statistics for layer normalization (Ba

et al., 2016) are independent of other samples in the batch. However, the normalization of inputs is done across the features, resulting in different samples being normalized differently depending on its features. Layer normalization was initially used in the work of Gulrajani et al. (2017).

From a representation standpoint, the most effective method proposed up-to-date has been spectral normalization (Miyato et al., 2018). Introduced initially in the Spectral Normalization GAN (SN-GAN), spectral normalization normalizes the weights of the discriminator, contributing to a more stable training. Spectral normalization constrains the spectral norm of the discriminator’s layers, which imposes a Lipschitz condition in a more efficient and adaptable way.

4.3 Architectural Features and Training Procedures

In addition to the regularization and normalisation techniques described above, in recent years the community devised a number of heuristic techniques which have shown to further improve the stability of the training. These include:

Feature matching (Salimans et al., 2016) — Proposed to prevent the generator from overtraining on the given discriminator, feature matching requires the generator to not only maximize the probability of the generated sample being classified as real by the discriminator, but also account for the generated data to match with the target distribution. This is achieved by training the generator on the intermediate layer of the discriminator.

Minibatch discrimination (Salimans et al., 2016) — A method aimed at hindering mode collapse, minibatch discrimination offers a method for penalizing strong similarity between generated samples. By adding a layer in the discriminator, minibatch discrimination is able to compute the cross-sample distance and measure the diversity of generated data. The discriminator still outputs a single probability, but offers the data computed with the use of minibatch as side information.

Historical averaging (Salimans et al., 2016) — Designed to avoid the oscillating and cyclical behavior of GANs, historical averaging incorporates past values of parameters into the player’s cost function by adding a term $\| \theta - \frac{1}{t} \sum_{i=1}^t \theta[i] \|^2$, with $\theta[i]$ being the value of the parameters at past time i .

One-sided label smoothing (Salimans et al., 2016) — Hindering the discriminator from becoming overconfident and squashing generator’s gradients, one-sided label smoothing restricts the discriminator to output the probabilities between 0.1 and 0.9, instead of 0 and 1, thus handicapping the discriminator and alleviating the vanishing gradient problem.

Noise injection (Arjovsky et al., 2017) — Designed to tackle the training instability problem, in particular, vanishing gradients problem, injecting continuous noise to the inputs of the discriminator leads to smoothing the distribution of the probability mass, thus reducing the problem of non-overlapping support, problematic especially for standard GAN. Noise injection has been shown to be computationally efficient

and practical (Jenni and Favaro, 2019), however, leading to a lower quality of the generated image (Roth et al., 2017).

5. Open Problems

Based on the surveyed methods, this section highlights several open problems which may provide promising research directions towards stabilizing GAN training.

5.1 Combining Training Stabilization Methods

With the number of approaches coming from different disciplines and perspectives proposed in recent years, it is unclear how these techniques align with each other and how combinations affect the overall performance. Thus, an interesting avenue of research is the investigation of how these methods could be potentially applied jointly. An example of this idea is BigGAN (Brock et al., 2018), which achieved state-of-the-art results by combining already-known methods and subsequently scaling them. A proper combination of training techniques, architectural changes, and loss function modifications could prove to yield superior results. This is especially important in the context of game theory and gradient-based GAN variants, which have been consistently omitted in the architectural and loss function GAN literature (Hong et al., 2019; Wang et al., 2019; Creswell et al., 2018). Although these type of GANs often possess significant limitations, the methods outlined by them such as fictitious play or no-regret algorithms could provide an additional advantage in stabilizing the GAN training procedure.

5.2 Actor-Critic Methods for GAN Training

The link between GANs and reinforcement learning has already been described (Hong et al., 2019; Pfau and Vinyals, 2017; Finn et al., 2016); in particular, Pfau and Vinyals (2017) discussed the connection between GANs and actor-critic methods. However, numerous possible connections remain unexplored. One of the potentially beneficial stabilization technique known to improve the stability in actor-critic methods is the use of replay buffers, enabling off-policy updates based on the action chosen. Although not directly applicable, an extension of replay buffers, prioritized experience replay (Schaul et al., 2015) could serve as a conceptual inspiration for a different method for identifying and improving the networks on the outliers, with outliers being the samples with large discrepancy between real and generated samples.

5.3 Discriminator Generalization

GAN training relies on the idea of the discriminator producing useful feedback to the generator. This, however, is often hard to attain. Furthermore, it is not entirely possible to evaluate the quality of the discriminator’s feedback and performance solely on the basis of its loss (Thanh-Tung et al., 2017). For example, a theoretically robust discriminator could have in practice learnt to discern real from generated images only based on a small set of features with no proper generalization. A crucial aspect of the optimal discriminator is its ability to generalize to unseen samples. A proper GAN objective should induce the discriminator towards the optimal discriminator with good generalization capabilities. This issue has

been explored more in-depth only recently, proposing new distances (Zhang et al., 2018c; Arora et al., 2017) and regularization techniques (Thanh-Tung et al., 2017). Nevertheless, constructing a GAN objective that fully incorporates the discriminator’s generalization capabilities remains a challenge.

5.4 Opponent Modeling in GANs

The two-player zero-sum game is a subject of rich literature in the field of opponent modeling (Albrecht and Stone, 2018). Nevertheless, very few studies have explored the connection between the GAN game and opponent modeling or multi-agent learning broadly. An interesting avenue would be to model the GAN game as a game between two agents with limited information and devise mechanisms that would allow to incorporate the awareness of the other agent into their training procedure. Such an approach could potentially lead to more stable convergence (Schäfer et al., 2019).

5.5 Variational Autoencoder GAN Hybrid

Similar to GANs, Variational Autoencoders (VAEs) are also a type of generative model (Kingma and Welling, 2014). However, in contrast to the former, which implicitly model a target distribution, VAEs follow an explicit approach to approximately estimating the data distribution through the use of an encoder and a decoder. VAEs are known to produce images of lesser quality in comparison to GANs. Nevertheless, they possess stronger theoretical groundings and have been shown to be less prone to mode collapse than GANs. Drawing on VAEs, VAEGAN (Larsen et al., 2016) and α -GAN (Rosca et al., 2017) combined the approaches to alleviate the mode collapse problem. This area at the intersection of VAEs and GANs, although not yet entirely explored, provides promising results and could prove to achieve significant results, especially in terms of preventing mode collapse.

6. Conclusion

This article surveyed current approaches for stabilizing the GAN training procedure, categorizing and outlining various techniques and key concepts. We contributed a comparative summary of methods together with a description of potential research directions to address open problems. We conclude that although recent years have seen an increase of published studies providing a number of methods to overcoming GAN training instability, the majority of these studies are limited to preventing only one training issue and often lack rigorous theoretical justification.

References

- Stefano V. Albrecht and Peter Stone. Autonomous agents modelling other agents: A comprehensive survey and open problems. *Artificial Intelligence*, 258:66–95, 2018. ISSN 00043702. doi: 10.1016/j.artint.2018.01.002.
- Isabela Albuquerque, João Monteiro, Thang Doan, Breandan Considine, Tiago Falk, and Ioannis Mitliagkas. Multi-objective training of Generative Adversarial Networks with multiple discriminators. 2019. URL <http://arxiv.org/abs/1901.08680>.

- Martin Arjovsky and Léon Bottou. Towards Principled Methods for Training Generative Adversarial Networks. In *International Conference on Learning Representation (ICLR 2017)*, pages 1–17, 2017.
- Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein GAN. 2017. URL <http://arxiv.org/abs/1701.07875>.
- Sanjeev Arora, Rong Ge, Yingyu Liang, Tengyu Ma, and Yi Zhang. Generalization and Equilibrium in Generative Adversarial Nets. pages 1–27, 2017.
- Sanjeev Arora, Andrej Risteski, and Yi Zhang. Do GANs Learn the Distribution? Some Theory and Empirics. pages 1–16, 2018.
- Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. Layer Normalization. 2016. URL <http://arxiv.org/abs/1607.06450>.
- David Balduzzi, Sebastien Racaniere, James Martens, Jakob Foerster, Karl Tuyls, and Thore Graepel. The Mechanics of n-Player Differentiable Games. 2018. URL <http://arxiv.org/abs/1802.05642>.
- S A Barnett. *Convergence Problems with Generative Adversarial Networks (GANs)*. PhD thesis, 2018.
- Marc G. Bellemare, Ivo Danihelka, Will Dabney, Shakir Mohamed, Balaji Lakshminarayanan, Stephan Hoyer, and Rémi Munos. The Cramer Distance as a Solution to Biased Wasserstein Gradients. pages 1–20, 2017. URL <http://arxiv.org/abs/1705.10743>.
- David Berthelot, Thomas Schumm, and Luke Metz. BEGAN : Boundary Equilibrium Generative Adversarial Networks. pages 1–10, 2017.
- Andrew Brock, Jeff Donahue, and Karen Simonyan. Large Scale GAN Training for High Fidelity Natural Image Synthesis. In *International Conference on Learning Representation (ICLR 2019)*, pages 1–35, 2018. URL <http://arxiv.org/abs/1809.11096>.
- Xi Chen, Yan Duan, Rein Houthoofd, John Schulman, Ilya Sutskever, and Pieter Abbeel. InfoGAN : Interpretable Representation Learning by Information Maximizing Generative Adversarial Nets. In *Advances in Neural Information Processing Systems*, number Nips, 2016.
- Corinna Cortes and Vladimir Vapnik. Support-Vector Networks. *Machine Learning*, 1995. ISSN 15730565. doi: 10.1023/A:1022627411411.
- Antonia Creswell, Tom White, Vincent Dumoulin, Kai Arulkumaran, Biswa Sengupta, and Anil A Bharath. Generative adversarial networks: An overview. *IEEE Signal Processing Magazine*, 35(1):53–65, 2018.
- Constantinos Daskalakis, Andrew Ilyas, Vasilis Syrgkanis, and Haoyang Zeng. Training GANs with Optimism. pages 1–30, 2017. URL <http://arxiv.org/abs/1711.00141>.

- Emily Denton, Soumith Chintala, Arthur Szlam, and Rob Fergus. Deep generative image models using a Laplacian pyramid of adversarial networks. In *Advances in neural information processing systems*, pages 1486–1494, 2015.
- Chris Donahue, Julian McAuley, and Miller Puckette. Synthesizing Audio with Generative Adversarial Networks. In *International Conference on Learning Representation (ICLR 2019)*, 2019.
- Ishan Durugkar, Ian Gemp, Sridhar Mahadevan, and Computer Sciences. Generative Multi-Adversarial Networks. In *International Conference on Learning Representation (ICLR 2017)*, pages 1–14, 2017.
- William Fedus, Mihaela Rosca, and Balaji Lakshminarayanan. Many Paths to Equilibrium: GANs do not need to decrease a divergence at every step. In *International Conference on Learning Representation (ICLR 2018)*, pages 1–21, 2018.
- Chelsea Finn, Paul Christiano, Pieter Abbeel, and Sergey Levine. A connection between generative adversarial networks, inverse reinforcement learning and Energy-based Models. 2016.
- Yoav Freund and Robert E Schapire. Adaptive Game Playing Using Multiplicative Weights. 103:79–103, 1999.
- Hao Ge, Yin Xia, Xu Chen, Randall Berry, and Ying Wu. Fictitious GAN: Training GANs with Historical Models. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 11205 LNCS(1): 122–137, 2018. ISSN 16113349. doi: 10.1007/978-3-030-01246-5.8.
- Arnab Ghosh, Viveka Kulharia, and Vinay Namboodiri. Message Passing Multi-Agent GANs. (2016), 2016. URL <http://arxiv.org/abs/1612.01294>.
- Arnab Ghosh, Viveka Kulharia, Vinay Namboodiri, Philip H S Torr, and Puneet Dokania. Multi-Agent Diverse Generative Adversarial Networks. In *CVPR*, 2017.
- Ian Goodfellow. NIPS 2016 Tutorial: Generative Adversarial Networks. *Advances in Neural Information Processing Systems*, 2016.
- Ian J. Goodfellow. On distinguishability criteria for estimating generative models. In *International Conference on Learning Representation (ICLR 2015)*, pages 1–6, 2015. URL <http://arxiv.org/abs/1412.6515>.
- Ian J Goodfellow, Jean Pouget-abadie, Mehdi Mirza, Bing Xu, and David Warde-farley. Generative Adversarial Nets. In *Advances in Neural Information Processing Systems*, pages 1–9, 2014.
- Paulina Grnarova, Kfir Y Levy, Thomas Hofmann, and Andreas Krause. An Online Learning Approach to Generative Adversarial Networks. pages 1–27, 2017.
- Paulina Grnarova, Kfir Y Levy, Aurelien Lucchi, Nathanael Perraudin, Thomas Hofmann, and Andreas Krause. Evaluating GANs via Duality. pages 1–16, 2018.

- Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron Courville. Improved Training of Wasserstein GANs. 2017.
- Elad Hazan, Karan Singh, and Cyril Zhang. Efficient Regret Minimization in Non-Convex Games. pages 1–18, 2017. URL <http://arxiv.org/abs/1708.00075>.
- He He, Jordan Boyd-Graber, Kevin Kwok, and Hal Daumé. Opponent Modeling in Deep Reinforcement Learning. 48, 2016. URL <http://arxiv.org/abs/1609.05559>.
- Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium. In *Advances in Neural Information Processing Systems*, number Nips, 2017.
- Saifuddin Hitawala. Comparative Study on Generative Adversarial Networks. 2018. URL <http://arxiv.org/abs/1801.04271>.
- Quan Hoang, Tu Dinh Nguyen, Trung Le, and Dinh Phung. MGAN: Training Generative Adversarial Nets with Multiple Generators. pages 1–23, 2017.
- Yongjun Hong, Uiwon Hwang, Jaeyoon Yoo, and Sungroh Yoon. How Generative Adversarial Networks and Their Variants Work. *ACM Computing Surveys*, 52(1):1–43, 2019. ISSN 03600300. doi: 10.1145/3301282.
- Ya-Ping Hsieh, Chen Liu, and Volkan Cevher. Finding Mixed Nash Equilibria of Generative Adversarial Networks. pages 1–30, 2018. URL <http://arxiv.org/abs/1811.02002>.
- Xan Huang, Yixuan Li, Omid Poursaeed, John Hopcroft, and Serge Belongie. Stacked Generative Adversarial Networks. 2017.
- F Huszar. How (not) to train your Generative Model: Scheduled Sampling, Likelihood, Adversary? pages 1–9, 2016.
- Stephanie L. Hyland, Cristóbal Esteban, and Gunnar Rätsch. Real-valued (Medical) Time Series Generation with Recurrent Conditional GANs. 2017. URL <http://arxiv.org/abs/1706.02633>.
- Daniel Im Jiwoong, He Ma, Chris Dongjoo Kim, and Graham Taylor. Generative Adversarial Parallelization. pages 1–20, 2016. URL <http://arxiv.org/abs/1612.04021>.
- Sergey Ioffe and Christian Szegedy. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. *Journal of Machine Learning Research*, 10(6):730–743, 2016. ISSN 17512794. doi: 10.1080/17512786.2015.1058180.
- Simon Jenni and Paolo Favaro. On Stabilizing Generative Adversarial Training with Noise. In *CVPR*, 2019. URL <http://arxiv.org/abs/1906.04612>.
- Alexia Jolicoeur-Martineau. GANs beyond divergence minimization. (1):1–14, 2018. URL <http://arxiv.org/abs/1809.02145>.

- Alexia Jolicoeur-martineau. The Relativistic Discriminator: A key element missing from Standard GAN. In *International Conference on Learning Representation (ICLR 2019)*, number 2017, pages 1–26, 2019.
- Felix Juefei-Xu, Vishnu Naresh Boddeti, and Marios Savvides. Gang of GANs: Generative Adversarial Networks with Maximum Margin Ranking. 2017. URL <http://arxiv.org/abs/1704.04865>.
- Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive Growing of GANs for improved quality, stability, and variation. In *International Conference on Learning Representation (ICLR 2018)*, pages 1–26, 2018.
- Diederik P Kingma and Max Welling. Auto-encoding variational Bayes. *International Conference on Learning Representations*, 2014.
- Naveen (Georgia Institute of Technology) Kodali, Jacob Abernethy, James Hayes, and Zsolt Kira. On Convergence and Stability of GANs. 2017.
- Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. ImageNet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*, 2012. ISBN 9781627480031.
- Karol Kurach, Mario Lucic, Xiaohua Zhai, Marcin Michalski, and Sylvain Gelly. A Large-Scale Study on Regularization and Normalization in GANs. 2018. URL <http://arxiv.org/abs/1807.04720>.
- Anders Boesen Lindbo Larsen, Søren Kaae Sønderby, Hugo Larochelle, and Ole Winther. Autoencoding beyond pixels using a learned similarity metric. In *33rd International Conference on Machine Learning, ICML 2016*, 2016. ISBN 9781510829008.
- Alistair Letcher, David Balduzzi, Sebastien Racaniere, James Martens, Jakob Foerster, Karl Tuyls, and Thore Graepel. Differentiable Game Mechanics. 20:1–40, 2019. URL <http://arxiv.org/abs/1905.04926>.
- Chun Liang Li, Wei Cheng Chang, Yu Cheng, Yiming Yang, and Barnabás Póczos. MMD GAN: Towards deeper understanding of moment matching network. *Advances in Neural Information Processing Systems*, 2017-Decem(MMD):2204–2214, 2017. ISSN 10495258.
- Yujia Li, Kevin Swersky, and Richard Zemel. Generative moment matching networks. In *32nd International Conference on Machine Learning, ICML 2015*, 2015. ISBN 9781510810587.
- Jae Hyun Lim and Jong Chul Ye. Geometric GAN. (Mmd):1–17, 2017. URL <http://arxiv.org/abs/1705.02894>.
- Ryan Lowe, Yi Wu, Aviv Tamar, Jean Harb, Pieter Abbeel, and Igor Mordatch. Multi-Agent Actor-Critic for Mixed Cooperative-Competitive Environments. 2017. URL <http://arxiv.org/abs/1706.02275>.

- Mario Lucic, Karol Kurach, Marcin Michalski, Olivier Bousquet, and Sylvain Gelly. Are GANs Created Equal? A Large-Scale Study. In *Advances in Neural Information Processing Systems*, number NIPS, 2018.
- Xudong Mao, Qing Li, Raymond Y K Lau, Zhen Wang, Haoran Xie, and Stephen Paul Smolley. Least Squares Generative Adversarial Networks. In *ICCV*, pages 2794–2802, 2016.
- Panayotis Mertikopoulos, Christos Papadimitriou, and Georgios Piliouras. Cycles in adversarial regularized learning. pages 1–22, 2017.
- Lars Mescheder, Sebastian Nowozin, and Andreas Geiger. The Numerics of GANs. In *Advances in Neural Information Processing Systems*, number Nips, 2017.
- Lars Mescheder, Andreas Geiger, and Sebastian Nowozin. Which Training Methods for GANs do actually Converge? 2018.
- Luke Metz, Ben Poole, David Pfau, and Jascha Sohl-Dickstein. Unrolled Generative Adversarial Networks. In *International Conference on Learning Representation (ICLR 2017)*, pages 1–25, 2017.
- Takeru Miyato, Toshiki Kataoka, Masanori Koyama, Yuichi Yoshida, and Preferred Networks. Spectral Normalization for Generative Adversarial Networks. 2018.
- Youssef Mroueh and Tom Sercu. Fisher GAN. In *Advances in Neural Information Processing Systems*, number Nips, 2017.
- John Nash. Non-Cooperative Games. *The Annals of Mathematics*, 1951. ISSN 0003486X. doi: 10.2307/1969529.
- Tu Dinh Nguyen, Trung Le, Hung Vu, and Dinh Phung. Dual Discriminator Generative Adversarial Nets. In *Advances in Neural Information Processing Systems*, number Nips, 2017.
- Sebastian Nowozin, Botond Cseke, and Ryota Tomioka. f-GAN: Training Generative Neural Samplers using Variational Divergence Minimization. 13(1):87–88, 2016. ISSN 00224405. doi: 10.1016/0022-4405(75)90045-x.
- Augustus Odena, Christopher Olah, and Jonathon Shlens. Conditional image synthesis with auxiliary classifier gans. *34th International Conference on Machine Learning, ICML 2017*, 6:4043–4055, 2017.
- Frans A. Oliehoek, Edwin D. De Jong, and Nikos Vlassis. The parallel nash memory for asymmetric games. In *GECCO 2006 - Genetic and Evolutionary Computation Conference*, 2006. ISBN 1595931864.
- Frans A. Oliehoek, Rahul Savani, Jose Gallego-Posada, Elise van der Pol, Edwin D. de Jong, and Roderich Gross. GANGs: Generative Adversarial Network Games. 2017. URL <http://arxiv.org/abs/1712.00679>.

- Frans A. Oliehoek, Rahul Savani, Jose Gallego, Elise van der Pol, and Roderich Groß. Beyond Local Nash Equilibria for Adversarial Networks. 2018. URL <http://arxiv.org/abs/1806.07268>.
- Georgios Papoudakis, Filippos Christianos, Arrasy Rahman, and Stefano V Albrecht. Dealing with non-stationarity in multi-agent deep reinforcement learning. *arXiv preprint arXiv:1906.04737*, 2019.
- David Pfau and Oriol Vinyals. Connecting Generative Adversarial Networks and Actor-Critic Methods. 2017.
- Ben Poole, Alexander A. Alemi, Jascha Sohl-Dickstein, and Anelia Angelova. Improved generator objectives for GANs. 2016. URL <http://arxiv.org/abs/1612.02780>.
- Guo-jun Qi. Loss-Sensitive Generative Adversarial Networks on Lipschitz. pages 1–34, 2018.
- Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised Representation Learning with Deep Convolutional Adversarial Networks. In *International Conference on Learning Representation (ICLR 2016)*, pages 1–16, 2016.
- Mihaela Rosca, Balaji Lakshminarayanan, David Warde-Farley, and Shakir Mohamed. Variational Approaches for Auto-Encoding Generative Adversarial Networks. *arXiv:1509.06461 [cs]*, 2017. URL <http://arxiv.org/abs/1706.04987>.
- Kevin Roth, Sebastian Nowozin, and Thomas Hofmann. Stabilizing Training of Generative Adversarial Networks through Regularization. 1(2):1–11, 2017.
- Yunus Saatchi and Andrew Gordon Wilson. Bayesian GAN. *Advances in Neural Information Processing Systems*, 2017-Decem(Nips):3623–3632, 2017. ISSN 10495258.
- Tim Salimans, Ian Goodfellow, Vicki Cheung, Alec Radford, and Xi Chen. Improved Techniques for Training GANs. pages 1–10, 2016.
- Florian Schäfer and Anima Anandkumar. Competitive gradient descent. In *Advances in Neural Information Processing Systems*, pages 7623–7633, 2019.
- Florian Schäfer, Hongkai Zheng, and Anima Anandkumar. Implicit competitive regularization in GANs. pages 1–10, 2019. URL <http://arxiv.org/abs/1910.05852>.
- Tom Schaul, John Quan, Ioannis Antonoglou, and David Silver. Prioritized Experience Replay. pages 1–21, 2015. URL <http://arxiv.org/abs/1511.05952>.
- Arunesh Sinha, Fei Fang, Bo An, Christopher Kiekintveld, and Milind Tambe. Stackelberg security games: Looking beyond a decade of success. In *IJCAI International Joint Conference on Artificial Intelligence*, 2018. ISBN 9780999241127.
- Bharath K. Sriperumbudur, Kenji Fukumizu, Arthur Gretton, Bernhard Schölkopf, and Gert R. G. Lanckriet. On integral probability metrics, ϕ -divergences and binary classification. (1):1–18, 2009. URL <http://arxiv.org/abs/0901.2698>.

- Hoang Thanh-Tung, Truyen Tran, and Svetha Venkatesh. Improving Generalization and Stability of Generative Adversarial Networks. pages 1–10, 2017.
- Lucas Theis, Aäron van den Oord, and Matthias Bethge. A note on the evaluation of generative models. pages 1–10, 2015. URL <http://arxiv.org/abs/1511.01844>.
- Ilya Tolstikhin, Sylvain Gelly, Olivier Bousquet, Carl Johann Simon-Gabriel, and Bernhard Schölkopf. AdaGAN: Boosting generative models. *Advances in Neural Information Processing Systems*, 2017-Decem:5425–5434, 2017. ISSN 10495258.
- Thomas Unterthiner, Bernhard Nessler, Calvin Seward, Günter Klambauer, Martin Heusel, Hubert Ramsauer, and Sepp Hochreiter. Coulomb GANs: Provably Optimal Nash Equilibrium via Potential Fields. pages 1–21, 2018.
- John von Neumann and Oskar Morgenstern. *Theory of games and economic behavior*. 2007. ISBN 0691119937.
- Chaoyue Wang, Chang Xu, Xin Yao, and Dacheng Tao. Evolutionary Generative Adversarial Networks. *2018*, pages 1–14.
- Kunfeng Wang, Chao Gou, Yanjie Duan, Yilun Lin, Xinhua Zheng, and Fei-Yue Wang. Generative adversarial networks: introduction and outlook. *IEEE/CAA Journal of Automatica Sinica*, 4(4):588–598, 2017a.
- Ruohan Wang, Antoine Cully, Hyung Jin Chang, Yiannis Demiris, and United Kingdom. MAGAN : Margin Adaptation for Generative Adversarial Networks. 2017b.
- Zhengwei Wang, Qi She, and Tomas E. Ward. Generative Adversarial Networks: A Survey and Taxonomy. (2):1–16, 2019. URL <http://arxiv.org/abs/1906.01529>.
- Lantao Yu, Weinan Zhang, Jun Wang, and Yong Yu. SeqGAN : Sequence Generative Adversarial Nets with Policy Gradient. (Goodfellow):2852–2858, 2016.
- Guanghao Zhang, Enmei Tu, and Dongshun Cui. Stable and improved generative adversarial nets (GANS): A constructive survey. *Proceedings - International Conference on Image Processing, ICIP*, 2017-Sept:1871–1875, 2018a. ISSN 15224880. doi: 10.1109/ICIP.2017.8296606.
- Han Zhang, Ian Goodfellow, Dimitris Metaxas, and Augustus Odena. Self-Attention Generative Adversarial Networks. 2019. URL <http://arxiv.org/abs/1805.08318>.
- Hongyang Zhang, Susu Xu, Jiantao Jiao, Pengtao Xie, Ruslan Salakhutdinov, Eric Xing, and Eric P Xing. Stackelberg GAN : Towards Provable Minimax Equilibrium via. pages 1–27, 2018b.
- Pengchuan Zhang, Qiang Liu, Dengyong Zhou, Tao Xu, and Xiaodong He. On the Discrimination-Generalization trade-off in GANs. In *International Conference on Learning Representation (ICLR 2018)*, volume 1, pages 1–25, 2018c. URL <http://arxiv.org/abs/1809.02145>.

- Junbo Zhao, Michael Mathieu, Yann Lecun, and Facebook Artificial. Energy-based Generative Adversarial Networks. In *International Conference on Learning Representation (ICLR 2017)*, number 2014, pages 1–17, 2017.
- Jun Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. Unpaired Image-to-Image Translation Using Cycle-Consistent Adversarial Networks. In *Proceedings of the IEEE International Conference on Computer Vision*, 2017. ISBN 9781538610329. doi: 10.1109/ICCV.2017.244.