

# CRYSTALGAN: LEARNING TO DISCOVER CRYSTALLOGRAPHIC STRUCTURES WITH GENERATIVE ADVERSARIAL NETWORKS

**Anonymous authors**

Paper under double-blind review

## ABSTRACT

Our main motivation is to propose an efficient approach to generate novel multi-element stable chemical compounds that can be used in real world applications. This task can be formulated as a combinatorial problem, and it takes many hours of human experts to construct, and to evaluate new data. Unsupervised learning methods such as Generative Adversarial Networks (GANs) can be efficiently used to produce new data. Cross-domain Generative Adversarial Networks were reported to achieve exciting results in image processing applications. However, in the domain of materials science, there is a need to synthesize data with higher order complexity compared to observed samples, and the state-of-the-art cross-domain GANs can not be adapted directly. In this contribution, we propose a novel GAN called CrystalGAN which generates new chemically stable crystallographic structures with increased domain complexity. We introduce an original architecture, we provide the corresponding loss functions, and we show that the CrystalGAN generates very reasonable data. We illustrate the efficiency of the proposed method on a real original problem of novel hydrides discovery that can be further used in development of hydrogen storage materials.

## 1 INTRODUCTION

In modern society, a big variety of inorganic compositions are used for hydrogen storage owing to its favorable cost (Crivello et al., 2016). A vast number of organic molecules are applied in solar cells, organic light-emitting diodes, conductors, and sensors (Yang et al., 2017). Synthesis of new organic and inorganic compounds is a challenge in physics, chemistry and in materials science. Design of new structures aims to find the best solution in a big chemical space, and it is in fact a combinatorial optimization problem.

In this work, we focus on applications of hydrogen storage, and in particular, we challenge the problem to investigate novel chemical compositions with stable crystals. Traditionally, density functional theory (DFT) plays a central role in prediction of chemically relevant compositions with stable crystals (Seko et al., 2018). However, the DFT calculations are computationally expensive, and it is not acceptable to apply it to test all possible randomly generated structures.

A number of machine learning approaches were proposed to facilitate the search for novel stable compositions (Butler et al., 2018). There was an attempt to find new compositions using an inorganic crystal structure database, and to estimate the probabilities of new candidates based on compositional similarities. These methods to generate relevant chemical compositions are based on recommender systems (Hu et al., 2008). The output of the recommender systems applied in the crystallographic field is a rating or preference for a structure. A recent approach based on a combination of machine learning methods and the high-throughput DFT calculations allowed to explore ternary chemical compounds (Schmidt et al., 2018), and it was shown that statistical methods can be of a big help to identify stable structures, and that they do it much faster than standard methods. Recently, support vector machines were tested to predict crystal structures (Oliynyk et al., 2017) showing that the method can reliably predict the crystal structure given its composition. It is worth mentioning that data representation of observations to be passed to a learner, is critical, and data

representations which are the most suitable for learning algorithms, are not necessarily scientifically intuitive (Swann et al., 2018).

Deep learning methods were reported to learn rich hierarchical models over all kind of data, and the GANs (Goodfellow et al., 2014) is a state-of-the-art model to synthesize data. Moreover, deep networks were reported to learn transferable representations (Ren & Lee, 2017). The GANs were already exploited with success in cross-domain learning applications for image processing (Zhu et al., 2017; Kim et al., 2017; Janz et al., 2017).

*Our goal is to develop a competitive approach to identify stable ternary chemical compounds, i.e., compounds containing three different elements, from observations of binary compounds.* Nowadays, there does not exist any approach that can be applied directly to such an important task of materials science. The state-of-the-art GANs are limited in the sense that they do not generate samples in domains with increased complexity, e.g., the application where we aim to construct crystals with three elements from observations containing two chemical elements only. An attempt to learn many-to-many mappings was recently introduced by Almahairi et al. (2018), however, this promising approach does not allow to generate data of a higher-order dimension.

Our contribution is multi-fold:

- To our knowledge, we are the first to introduce a GAN to solve the scientific problem of discovery of novel crystal structures, and we introduce an original methodology to generate new stable chemical compositions;
- The proposed method is called CrystalGAN, and it consists of two cross-domain GAN blocks with constraints integrating prior knowledge including a feature transfer step;
- The proposed model generates data with increased complexity with respect to observed samples;
- We demonstrate by numerical experiments on a real challenge of chemistry and materials science that our approach is competitive compared to existing methods;
- The proposed algorithm is efficiently implemented in Python, and it will be publicly available shortly, as soon as the contribution is de-anonymized.

This paper is organized as follows. We discuss the related work in Section 2. In Section 3, we provide the formalisation of the problem, and introduce the CrystalGAN. The results of our numerical experiments are discussed in Section 4. Concluding remarks and perspectives close the paper.

## 2 RELATED WORK

Our contribution is closely related to the problems of unsupervised learning and cross-domain learning, since our aim is to synthesize novel data, and the new samples are supposed to belong to an unobserved domain with an augmented complexity.

In the adversarial nets framework, the deep generative models compete with an adversary which is a discriminative model learning to identify whether an observation comes from the model distribution or from the data distribution (Goodfellow, 2016). A classical GAN consists of two models, a generator  $\mathbf{G}$  whose objective is to synthesize data and a discriminator  $\mathbf{D}$  whose aim is to distinguish between real and generated data. The generator and the discriminator are trained simultaneously, and the training problem is formulated as a two-player minimax game. A number of techniques to improve training of GANs were proposed by Arjovsky et al. (2017); Gulrajani et al. (2017); Salimans et al. (2016).

Learning cross domain relations is an active research direction in image processing. Several recent papers (Zhu et al., 2017; Kim et al., 2017; Almahairi et al., 2018) discuss an idea to capture some particular characteristics of one image and to translate them into another image.

A conditional GAN for image-to-image translation is considered by Isola et al. (2017). An advantage of the conditional model is that it allows to integrate underlying structure into the model. The conditional GANs were also used for multi-model tasks (Mirza & Osindero, 2014). An idea to combine observed data to produce new data was proposed in (Yazdani, 2017), e.g., an artist can mix existing pieces of music to create a new one.

$AH$ :	First domain, H is hydrogen, and $A$ is a metal
$BH$ :	Second domain, H is hydrogen, and $B$ is another metal
$\mathbf{G}_{AHB_1}$ :	Generator function that translates input features $x_{AH}$ from (domain) $AH$ to $BH$
$\mathbf{G}_{BHA_1}$ :	Generator function that translates input features $x_{BH}$ from (domain) $BH$ to $AH$
$\mathbf{D}_{AH}$ and $\mathbf{D}_{BH}$ :	Discriminator functions of $AH$ domain and $BH$ domain, respectively
$AHB_1$ :	$x_{AHB_1}$ is a sample generated by generator function $\mathbf{G}_{AHB_1}$
$BHA_1$ :	$y_{BHA_1}$ is a sample produced by generator function $\mathbf{G}_{BHA_1}$
$AHBA_1$ and $BHAB_1$ :	Data reconstructed after two generator translations
$AHB_g$ and $BHA_g$ :	Data obtained after feature transfer step from domain $AH$ to domain $BH$ , and from domain $BH$ to domain $AH$ , respectively
	Input data for the second step of CrystalGAN
$\mathbf{G}_{AHB_2}$ :	Generator function that translates $x_{AHB_g}$ to $AHB_2$
	Features generated in the first step from $AHB_g$ to $AHB_2$
$\mathbf{G}_{BHA_2}$ :	Generator function that translates $y_{BHA_g}$ to $BHA_2$
	Data generated in first step from $BHA_g$ to $BHA_2$
$\mathbf{D}_{AHB}$ and $\mathbf{D}_{BHA}$ :	The discriminator functions of domain $AHB_g$ and domain $BHA_g$ , respectively
$AHB_2$ :	$x_{AHB_2}$ is a sample generated by the generator function $\mathbf{G}_{AHB_2}$
$BHA_2$ :	$y_{BHA_2}$ is a sample produced by the generator function $\mathbf{G}_{BHA_2}$
$AHBA_2$ and $BHAB_2$ :	Data reconstructed as a result of two generators translations
$AHB_2$ and $BHA_2$ :	Final new data (to be explored by human experts)

Table 1: Notations used in CrystalGAN.

An approach to learn high-level semantic features, and to train a model for more than a single task, was introduced by Ren & Lee (2017). In particular, it was proposed to train a model to jointly learn several complementary tasks. This method is expected to overcome the problem of overfitting to a single task. An idea to introduce multiple discriminators whose role varies from formidable adversary to forgiving teacher was discussed by Durugkar et al. (2017).

Several GANs were adapted to some materials science and chemical applications. So, Objective-Reinforced GANs that perform molecular generation of carbon-chain sequence taking into consideration some desired properties, were introduced in (Sanchez-Lengeling et al., 2017), and the method was shown to be efficient for drug discovery. Another avenue is to integrate rule-based knowledge, e.g., molecular descriptors with the deep learning. ChemNet (Goh et al., 2017) is a deep neural network pre-trained with chemistry-relevant representations obtained from prior knowledge. The model can be used to predict new chemical properties. However, as we have already mentioned before, none of these methods generates crystal data of augmented complexity.

### 3 CRYSTALGAN: AN APPROACH TO GENERATE STABLE TERNARY CHEMICAL COMPOUNDS

In this section, we introduce our approach. The CrystalGAN consists of three procedures:

1. First step GAN which is closely related to the cross-domain GANs, and that generates pseudo-binary samples where the domains are mixed.
2. Feature transfer procedure constructs higher order complexity data from the samples generated at the previous step, and where components from all domains are well-separated.
3. Second step GAN synthesizes, under geometric constraints, novel ternary stable chemical structures.

First, we describe a cross-domain GAN, and then, we provide all the details on the proposed CrystalGAN. We provide all notations used by the CrystalGAN in Table 1. The GANs architectures for the first and the second steps are shown on Figure 1.

#### 3.1 PROBLEM FORMULATION FOR APPLICATIONS WITH AUGMENTED COMPLEXITY

We now propose a novel architecture based on the cross-domain GAN algorithms with constraint learning to discover higher order complexity crystallographic systems. We introduce a GAN model to find relations between different crystallographic domains, and to generate new materials.

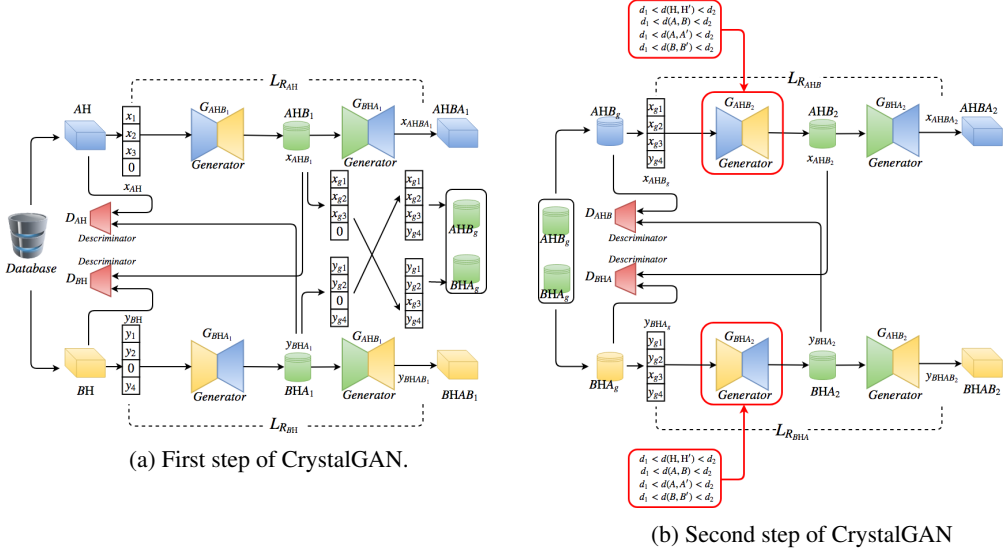


Figure 1: The CrystalGAN architecture.

To make the paper easier to follow, without loss of generality, we will present our method providing a specific example of generating ternary hydride compounds of the form "A (a metal) - H (hydrogen) - B (a metal)".

Our observations are stable binary compounds containing chemical elements  $A+H$  which is a composition of some metal  $A$  and the hydrogen  $H$ , and  $B+H$  which is a mixture of another metal  $B$  with the hydrogen. So, a machine learning algorithm has access to observations  $\{(x_{AH_i})\}_{i=1}^{N_{AH}}$  and  $\{(y_{BH_i})\}_{i=1}^{N_{BH}}$ . Our goal is to generate novel ternary, i.e. more complex, stable data  $x_{AHB}$  (or  $y_{BHA}$ ) based on the properties learned from the observed binary structures.

Below we describe the architecture of the CrystalGAN.

### 3.2 STEPS OF CRYSTALGAN

Our approach consists of two consecutive steps with a feature transfer procedure inbetween.

#### 3.2.1 FIRST STEP

The first step of CrystalGAN generates new data with increased complexity. The adversarial network takes  $\{(x_{AH_i})\}_{i=1}^{N_{AH}}$  and  $\{(y_{BH_i})\}_{i=1}^{N_{BH}}$ , and synthesizes

$$x_{AHB_1} = \mathbf{G}_{AHB_1}(x_{AH}), x_{AHBA_1} = \mathbf{G}_{BHA_1}(x_{AHB_1}) = \mathbf{G}_{BHA_1} \circ \mathbf{G}_{AHB_1}(x_{AH}), \quad (1)$$

and

$$y_{BHA_1} = \mathbf{G}_{BHA_1}(y_{BH}), y_{BHAB_1} = \mathbf{G}_{AHB_1}(y_{BHA_1}) = \mathbf{G}_{AHB_1} \circ \mathbf{G}_{BHA_1}(y_{BH}). \quad (2)$$

Figure 1a summarizes the first step of CrystalGAN.

The reconstruction loss functions take the following form:

$$\mathcal{L}_{RAH} = d(x_{AHBA_1}, x_{AH}) = d(\mathbf{G}_{BHA_1} \circ \mathbf{G}_{AHB_1}(x_{AH}), x_{AH}), \quad (3)$$

$$\mathcal{L}_{RBH} = d(y_{BHAB_1}, y_{BH}) = d(\mathbf{G}_{AHB_1} \circ \mathbf{G}_{BHA_1}(y_{BH}), y_{BH}). \quad (4)$$

Ideally,  $\mathcal{L}_{RAH} = 0$ ,  $\mathcal{L}_{RBH} = 0$ , and  $x_{AHBA_1} = x_{AH}$ ,  $y_{BHAB_1} = y_{BH}$ , and we minimize the distances  $d(x_{AHBA_1}, x_{AH})$  and  $d(y_{BHAB_1}, y_{BH})$ .

The generative adversarial loss functions of the first step of CrystalGAN aim to control that the original observations are reconstructed as accurate as possible:

$$\mathcal{L}_{GAN_{BH}} = -\mathbb{E}_{x_{AH} \sim P_{AH}} [\log(\mathbf{D}_{BH}(\mathbf{G}_{AHB_1}(x_{AH})))] , \quad (5)$$

and

$$\mathcal{L}_{\text{GAN}_{\text{AH}}} = -\mathbb{E}_{y_{\text{BH}} \sim P_{\text{BH}}} [\log(\mathbf{D}_{\text{AH}}(\mathbf{G}_{\text{BHA}_1}(y_{\text{BH}})))]. \quad (6)$$

The generative loss functions contain the two terms defined above:

$$\mathcal{L}_{\mathbf{G}_{\text{AHB}_1}} = \mathcal{L}_{\text{GAN}_{\text{BH}}} + \mathcal{L}_{R_{\text{AH}}}, \mathcal{L}_{\mathbf{G}_{\text{BHA}_1}} = \mathcal{L}_{\text{GAN}_{\text{AH}}} + \mathcal{L}_{R_{\text{BH}}}. \quad (7)$$

The discriminative loss functions aim to discriminate the samples coming from AH and BH:

$$\mathcal{L}_{\mathbf{D}_{\text{BH}}} = -\mathbb{E}_{y_{\text{BH}} \sim P_{\text{BH}}} [\log(\mathbf{D}_{\text{BH}}(y_{\text{BH}}))] - \mathbb{E}_{x_{\text{AH}} \sim P_{\text{AH}}} [\log(1 - \mathbf{D}_{\text{BH}}(\mathbf{G}_{\text{AHB}_1}(x_{\text{AH}})))], \quad (8)$$

$$\mathcal{L}_{\mathbf{D}_{\text{AH}}} = -\mathbb{E}_{x_{\text{AH}} \sim P_{\text{AH}}} [\log(\mathbf{D}_{\text{AH}}(x_{\text{AH}}))] - \mathbb{E}_{y_{\text{BH}} \sim P_{\text{BH}}} [\log(1 - \mathbf{D}_{\text{AH}}(\mathbf{G}_{\text{BHA}_1}(y_{\text{BH}})))]. \quad (9)$$

Now, we have all elements to define the full generative loss function of the first step:

$$\mathcal{L}_{\mathbf{G}_1} = \mathcal{L}_{\mathbf{G}_{\text{AHB}_1}} + \mathcal{L}_{\mathbf{G}_{\text{BHA}_1}} = \lambda_1 \mathcal{L}_{\text{GAN}_{\text{BH}}} + \lambda_2 \mathcal{L}_{R_{\text{AH}}} + \lambda_3 \mathcal{L}_{\text{GAN}_{\text{AH}}} + \lambda_4 \mathcal{L}_{R_{\text{BH}}}, \quad (10)$$

where  $\lambda_1, \lambda_2, \lambda_3,$  and  $\lambda_4$  are real-valued hyper-parameters that control the ratio between the corresponding terms, and the hyper-parameters are to be fixed by cross-validation.

The full discriminator loss function of this step  $\mathcal{L}_{\mathbf{D}_1}$  is defined as follows:

$$\mathcal{L}_{\mathbf{D}_1} = \mathcal{L}_{\mathbf{D}_{\text{AH}}} + \mathcal{L}_{\mathbf{D}_{\text{BH}}}. \quad (11)$$

### 3.2.2 FEATURE TRANSFER

The first step generates pseudo-binary samples  $MH$ , where  $M$  is a new discovered domain merging  $A$  and  $B$  properties. Although these results can be interesting for human experts, the samples generated by the first step are not easy to interpret, since the domains  $A$  and  $B$  are completely mixed in these samples, and there is no way to deduce characteristics of two separate elements coming from these domains.

So, we need a second step which will generate data of a higher order complexity from two given domains. We transfer the attributes of  $A$  and  $B$  elements, this procedure is also shown on Figure 1a, in order to construct a new dataset that will be used as a training set in the second step of the CrystalGAN.

In order to prepare the datasets to generate higher order complexity samples, we add a placeholder. (E.g., for domain AH, the fourth matrix is empty, and for domain BH, the third matrix is empty.)

### 3.2.3 SECOND STEP OF THE CRYSTALGAN

The second step GAN takes as input the data generated by the first step GAN and modified by the feature transfer procedure. The results of the second step are samples which describe ternary chemical compounds that are supposed to be stable from chemical viewpoint. *The geometric constraints control the quality of generated data.*

A crystallographic structure is fully described by a local distribution. This distribution is determined by distances to all nearest neighbors of each atom in a given crystallographic structure. We enforce the second step GAN with the following geometric constraints which satisfy the geometric conditions of our scientific domain application. The implemented constraints are also shown on Figure 1b.

Let  $\{s_i\}_{i=1}^m$  be the set of distances of the first neighbors of all atoms in a crystallographic structure. There are two geometric constraints to be considered while generating new data.

The first geometric (geo) constraint is defined as follows:

$$\mathcal{L}_{\text{geo}_1} = f(d_1, s_1, \dots, s_m) = \frac{1}{m} \sum_{i=1}^m \min_{i \in [1, m]} \|d_1 - s_i\|_2^2, \quad (12)$$

where  $d_1$  is the minimal distance between two first nearest neighbors in a given crystallographic structure.

The second geometric constraint takes the following form:

$$\mathcal{L}_{\text{geo}_2} = f(d_2, s_1, \dots, s_m) = -\frac{1}{m} \sum_{i=1}^m \min_{i \in [1, m]} \|d_2 - s_i\|_2^2, \quad (13)$$

where  $d_2$  is the maximal distance between two first nearest neighbors.

The loss function of the second step GAN is augmented by the following geometric constraints:

$$\mathcal{L}_{\text{geo}} = \mathcal{L}_{\text{geo}_1} + \mathcal{L}_{\text{geo}_2}. \quad (14)$$

Given  $x_{\text{AHB}_g}$  and  $y_{\text{BHA}_g}$  from the previous step, we generate:

$$x_{\text{AHB}_2} = \mathbf{G}_{\text{AHB}_2}(x_{\text{AHB}_g}), x_{\text{AHBA}_2} = \mathbf{G}_{\text{BHA}_2}(x_{\text{AHB}_2}) = \mathbf{G}_{\text{BHA}_2} \circ \mathbf{G}_{\text{AHB}_2}(x_{\text{AHB}_g}), \quad (15)$$

$$y_{\text{BHA}_2} = \mathbf{G}_{\text{BHA}_2}(y_{\text{BHA}_g}), y_{\text{BHAB}_2} = \mathbf{G}_{\text{AHB}_2}(y_{\text{BHA}_2}) = \mathbf{G}_{\text{AHB}_2} \circ \mathbf{G}_{\text{BHA}_2}(y_{\text{BHA}_g}). \quad (16)$$

The reconstruction loss functions are given:

$$\mathcal{L}_{R_{\text{AHB}}} = d(x_{\text{AHBA}_2}, x_{\text{AHB}_g}) = d(\mathbf{G}_{\text{BHA}_2} \circ \mathbf{G}_{\text{AHB}_2}(x_{\text{AHB}_g}), x_{\text{AHB}_g}), \quad (17)$$

$$\mathcal{L}_{R_{\text{BHA}}} = d(y_{\text{BHAB}_2}, y_{\text{BHA}_g}) = d(\mathbf{G}_{\text{AHB}_2} \circ \mathbf{G}_{\text{BHA}_2}(y_{\text{BHA}_g}), y_{\text{BHA}_g}). \quad (18)$$

The generative adversarial loss functions are given by:

$$\mathcal{L}_{\text{GAN}_{\text{BHA}_g}} = -\mathbb{E}_{x_{\text{AHB}_g} \sim P_{\text{AHB}_g}} [\log(\mathbf{D}_{\text{BHA}}(\mathbf{G}_{\text{AHB}_2}(x_{\text{AHB}_g})))] \quad (19)$$

$$\mathcal{L}_{\text{GAN}_{\text{AHB}_g}} = -\mathbb{E}_{y_{\text{BHA}_g} \sim P_{\text{BHA}_g}} [\log(\mathbf{D}_{\text{AHB}}(\mathbf{G}_{\text{BHA}_2}(y_{\text{BHA}_g})))] \quad (20)$$

The generative loss functions of the this step are defined as follows:

$$\mathcal{L}_{\mathbf{G}_{\text{AHB}_2}} = \mathcal{L}_{\text{GAN}_{\text{BHA}_g}} + \mathcal{L}_{R_{\text{AHB}}}, \quad (21)$$

$$\mathcal{L}_{\mathbf{G}_{\text{BHA}_2}} = \mathcal{L}_{\text{GAN}_{\text{AHB}_g}} + \mathcal{L}_{R_{\text{BHA}}}. \quad (22)$$

The losses of the discriminator of the second step can be defined:

$$\mathcal{L}_{\mathbf{D}_{\text{BHA}}} = -\mathbb{E}_{y_{\text{BHA}_g} \sim P_{\text{BHA}_g}} [\log(\mathbf{D}_{\text{BHA}}(y_{\text{BHA}_g}))] - \quad (23)$$

$$\mathbb{E}_{x_{\text{AHB}_g} \sim P_{\text{AHB}_g}} [\log(1 - \mathbf{D}_{\text{BHA}}(\mathbf{G}_{\text{AHB}_2}(x_{\text{AHB}_g})))] \quad (24)$$

$$\mathcal{L}_{\mathbf{D}_{\text{AHB}}} = -\mathbb{E}_{x_{\text{AHB}_g} \sim P_{\text{AHB}_g}} [\log(\mathbf{D}_{\text{AHB}}(x_{\text{AHB}_g}))] - \quad (25)$$

$$\mathbb{E}_{y_{\text{BHA}_g} \sim P_{\text{BHA}_g}} [\log(1 - \mathbf{D}_{\text{AHB}}(\mathbf{G}_{\text{BHA}_2}(y_{\text{BHA}_g})))] \quad (26)$$

Now, we have all elements to define the full generative loss function:

$$\mathcal{L}_{\mathbf{G}_2} = \mathcal{L}_{\mathbf{G}_{\text{AHB}_2}} + \mathcal{L}_{\mathbf{G}_{\text{BHA}_2}} + \mathcal{L}_{\text{geo}} = \quad (27)$$

$$\lambda_1 \mathcal{L}_{\text{GAN}_{\text{BHA}_g}} + \lambda_2 \mathcal{L}_{R_{\text{AHB}}} + \lambda_3 \mathcal{L}_{\text{GAN}_{\text{AHB}_g}} + \lambda_4 \mathcal{L}_{R_{\text{BHA}}} + \lambda_5 \mathcal{L}_{\text{geo}_1} + \lambda_6 \mathcal{L}_{\text{geo}_2}, \quad (28)$$

where  $\lambda_1, \lambda_2, \lambda_3, \lambda_4, \lambda_5,$  and  $\lambda_6$  are real-valued hyper-parameters that control the influence of the terms.

The full discriminative loss function of the second step  $\mathcal{L}_{\mathbf{D}_2}$  takes the form:

$$\mathcal{L}_{\mathbf{D}_2} = \mathcal{L}_{\mathbf{D}_{\text{AHB}}} + \mathcal{L}_{\mathbf{D}_{\text{BHA}}}. \quad (29)$$

To summarise, in the second step, we use the dataset issued from the feature transfer as an input containing two domains  $x_{\text{AHB}_g}$  and  $y_{\text{BHA}_g}$ . We train the cross-domain GAN taking into consideration constraints of the crystallographic environment. We integrated geometric constraints proposed by crystallographic and materials science experts to satisfy environmental constraints, and to increase the rate of synthesized stable ternary compounds. The second step is drafted on Figure 1b.

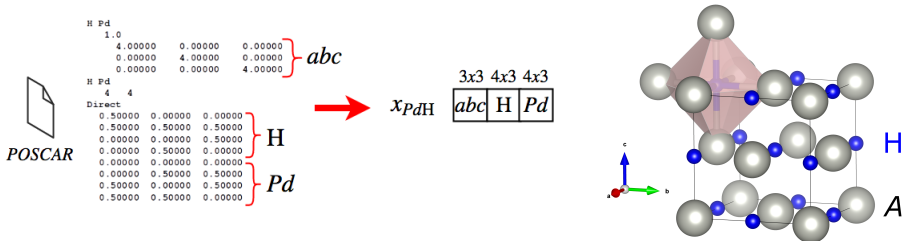


Figure 2: On the left: an example of a POSCAR file describing the composition of Palladium and Hydrogen, and the data representation in the CrystalGAN. On the right: a 3-dimensional representation.

## 4 EXPERIMENTS

### 4.1 TASK DESCRIPTION: EXPLORING NOVEL HYDRIDES

Crystallographic structures can be represented using the POSCAR files which are input files for the DFT calculations under the VASP code (Kresse & Joubert, 1999). These are coordinate files, they contain the lattice geometry and the atomic positions, as well as the number (or the composition) and the nature of atoms in the crystal unit cell.

We use a dataset constructed from (Bourgeois et al., 2017; Villars & Cenzual, 2017) by experts in materials science. Our training data set contains the POSCAR files, and the proposed CrystalGAN generates also POSCAR files. Such a file contains three matrices: the first one is  $abc$  matrix, corresponding to the three lattice vectors defining the unit cell of the system, the second matrix contains atomic positions of H atom, and the third matrix contains coordinates of metallic atom  $A$  (or  $B$ ). The information from the files is fed into 4-dimensional tensors. An example of a POSCAR file, and its corresponding representation for the GANs is shown on Figure 2. On the same figure on the right we show the corresponding structure in 3D. Note that we increase the data complexity by the feature transfer procedure by adding placeholders.

Our training dataset includes 1,416 POSCAR files of binary hydrides divided into 63 classes where each class is represented as a 4-dimensional tensor. Each class of binary  $MH$  hydride contains two elements: the hydrogen H and another element  $M$  from the periodic table.

In our experiments, after discussions with materials science researchers, we focused on exploration of ternary compositions "Palladium - Hydrogen - Nickel" from the binary systems observations of "Palladium - Hydrogen" and "Nickel - Hydrogen". So,  $AH = PdH$ , and  $BH = NiH$ . We also considered another task to generate ternary compounds "Magnesium - Hydrogen - Titanium".

In the CrystalGAN, we need to compute all the distances of the nearest neighbors for each generated POSCAR file. The distances between hydrogen atoms H in a given crystallographic structure should respect some geometric rules, as well as the distances between the atoms  $A-B$ ,  $A-A'$ , and  $B-B'$ . We applied the geometric constraints on the distances between the neighbors (for each atom in a crystallographic structure) introduced in the previous section. Note that the distances  $A-H$  and  $B-H$  are not penalized by the constraints.

### 4.2 IMPLEMENTATION DETAILS

In order to compute the distances between all nearest neighbors in the generated data, we used the pythonic library Pymatgen (Ong et al., 2012) specifically developed for material analysis.

For all experiments in this paper, the distances are fixed by our colleagues in crystallographic and materials science to  $d_1 = 1.8 \text{ \AA}$  (angstrom,  $10^{-10}$  meter) and  $d_2 = 3 \text{ \AA}$ . We set all the hyper-parameters by cross validation, however, we found that a reasonable performance is reached when all  $\lambda_i$  have similar values, and are quite close to 1. We use the standard AdamOptimizer with learning rate  $\alpha = 0.0001$ , and  $\beta_1 = 0.5$ . The number of epochs is set to 1000 (we verified that the functions converge). The mini-batch size equals 35.

Composition	GAN (standard)	DiscoGAN	CrystalGAN without constraints	CrystalGAN with constraints
Pd - Ni - H	0	0	4	9
Mg - Ti - H	0	0	2	8

Table 2: Number of ternary compositions of good quality generated by the tested methods.

Each block of the CrystalGAN architecture (the generators and the discriminators) is a multi-layer neural network with 5 hidden layers. Each layer contains 100 units. We use the rectified linear unit (ReLU) as an activation function of the neural network. All these parameters were fixed by cross-validation (for both chosen domains "Palladium - Hydrogen" and "Nickel - Hydrogen").

Our code is implemented in Python (TensorFlow). We run the experiments using GPU with graphics card NVIDIA Quadro M5000.

#### 4.3 RESULTS OF NUMERICAL EXPERIMENTS AND DISCUSSION

In our numerical experiments, we compare the proposed CrystalGAN with a classical GAN, the DiscoGAN (Kim et al., 2017), and the CrystalGAN but without the geometric constraints. All these GANs generate POSCAR files, and we *evaluate the performance of the models by the number of generated ternary structures which satisfy the geometric crystallographic environment*. Table 2 shows the number of successes for the considered methods. The classical GAN which takes Gaussian noise as an input, does not generate acceptable chemical structures. The DiscoGAN approach performs quite well if we use it to generate novel pseudo-binary structures, however, it is not adapted to synthesize ternary compositions. We observed that the CrystalGAN (with the geometric constraints) outperforms all tested methods.

From multiple discussions with experts in materials science and chemistry, first, we know that the number of novel stable compounds can not be very high, and it is already considered as a success if we synthesize several stable structures which satisfy the constraints. Hence, we can not really reason in terms of accuracy or error rate which are widely used metrics in machine learning and data mining.

Second, evaluation of a stable structure is not straightforward. Given a new composition, only the result of density functional theory (DFT) calculations can provide a conclusion whether this composition is stable enough, and whether it can be used in practice. However, the DFT calculations are computationally too expensive, and it is out of question to run them on all data we generated using the CrystalGAN. It is planned to run the DFT calculations on some pre-selected generated ternary compositions to take a final decision on practical utility of the chemical compounds.

## 5 CONCLUSIONS

Our goal was to develop a principled approach to generate new ternary stable crystallographic structures from observed binary, i.e. containing two chemical elements only. We propose a learning method called CrystalGAN to discover cross-domain relations in real data, and to generate novel structures. The proposed approach can efficiently integrate, in form of constraints, prior knowledge provided by human experts.

CrystalGAN is the first GAN developed to generate scientific data in the field of materials science. To our knowledge, it is also the first approach which generates data of a higher-order complexity, i.e., ternary structures where the domains are well-separated from observed binary compounds. The CrystalGAN was, in particular, successfully tested to tackle the challenge to discover new materials for hydrogen storage.

Currently, we investigate different GANs architectures, also including elements of reinforcement learning, to produce data even of a higher complexity, e.g., compounds containing four or five chemical elements. Note that although the CrystalGAN was developed and tested for applications in materials science, it is a general method where the constraints can be easily adapted to any scientific problem.



## REFERENCES

- A. Almahairi, S. Rajeshwar, A. Sordoni, P. Bachman, and A. Courville. Augmented cycleGAN: Learning many-to-many mappings from unpaired data. In *ICML*, 2018.
- Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein GAN. *arXiv:1701.07875*, 2017.
- N. Bourgeois, J.-C. Crivello, P. Cenedese, and J.-M. Joubert. Systematic first-principles study of binary metal hydrides. *ACS Combinatorial Science*, 19(8):513–523, 2017. doi: 10.1021/acscombsci.7b00050.
- K. T. Butler, D. W. Davies, H. Cartwright, O. Isayev, and A. Walsh. Machine learning for molecular and materials science. *Nature*, 559, 2018.
- J.-C. Crivello, B. Dam, R. V. Denys, M. Dornheim, D. M. Grant, J. Huot, T. R. Jensen, P. de Jongh, M. Latroche, C. Milanese, D. Milcius, G. S. Walker, C. J. Webb, C. Zlotea, and V. A. Yartys. Review of magnesium hydride-based materials: development and optimisation. *Applied Physics A*, 2016.
- Ishan Durugkar, Ian Gemp, and Sridhar Mahadevan. Generative Multi-Adversarial Networks. In *International Conference on Learning Representations (ICLR)*, 2017.
- Garrett B. Goh, Charles Siegel, Abhinav Vishnu, and Nathan Hodas. ChemNet: A Transferable and Generalizable Deep Neural Network for Small-molecule Property Prediction. In *NIPS Workshop on Machine Learning for Molecules and Materials*, 2017.
- Ian Goodfellow. NIPS 2016 Tutorial: Generative Adversarial Networks. *arxiv:1701.00160*, 2016.
- Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative Adversarial Nets. In *Advances in Neural Information Processing Systems 27 (NIPS)*, 2014.
- Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron Courville. Improved Training of Wasserstein GANs. In *Advances in Neural Information Processing Systems 30 (NIPS)*, 2017.
- Y. Hu, Y. Koren, and C. Volinsky. Collaborative Filtering for Implicit Feedback Datasets. In *ICDM*, 2008.
- Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. Image-to-Image Translation with Conditional Adversarial Networks. In *Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on*, 2017.
- David Janz, Jos van der Westhuizen, Brooks Paige, Matt J. Kusner, and José’ Miguel Hernández-Lobato. Learning a Generative Model for Validity in Complex Discrete Structures. In *NIPS Workshop on Machine Learning for Molecules and Materials*, 2017.
- Taeksoo Kim, Moonsu Cha, Hyunsoo Kim, Jung Kwon Lee, and Jiwon Kim. Learning to Discover Cross-Domain Relations with Generative Adversarial Networks. In *Proceedings of the 34th International Conference on Machine Learning*, 2017.
- G. Kresse and D. Joubert. From ultrasoft pseudopotentials to the projector augmented-wave method. *Physical Review B*, 1999.
- Mehdi Mirza and Simon Osindero. Conditional Generative Adversarial Nets. *arXiv:1411.1784*, 2014.
- Anton O. Oliynyk, Lawrence A. Adutwum, Brent W. Rudyk, Harshil Pisavadia, Sogol Lotfi, Viktor Hlukhyy, James J. Harynyuk, Arthur Mar, and Jakoah Brgoch. Disentangling Structural Confusion through Machine Learning: Structure Prediction and Polymorphism of Equiatomic Ternary Phases ABC. *Journal of the American Chemical Society*, 2017.

- Shyue Ping Ong, William Davidson Richards, Anubhav Jain, Geoffroy Hautier, Michael Kocher, Shreyas Cholia, Dan Gunter, Vincent L. Chevrier, Kristin A. Persson, and Gerbrand Ceder. Python Materials Genomics (pymatgen): A robust, open-source python library for materials analysis. *Computational Materials Science* 68 (2013) 314319, 2012.
- Z. Ren and Y. J. Lee. Cross-domain self-supervised multi-task feature learning using synthetic imagery. *ArXiv preprint arXiv:1711.09082*, 2017.
- Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved Techniques for Training GANs. In *Advances in Neural Information Processing Systems 29 (NIPS)*, 2016.
- Benjamin Sanchez-Lengeling, Carlos Outeiral, Gabriel L. Guimaraes, and Alán Aspuru-Guzik. Optimizing distributions over molecular space. An Objective-Reinforced Generative Adversarial Network for Inverse-design Chemistry (ORGANIC). *Preprint: chemrxiv:5309668*, 2017.
- Jonathan Schmidt, Liming Chen, Silvana Botti, and Miguel A. L. Marques. Predicting the stability of ternary intermetallics with density functional theory and machine learning. *Journal of Chemical Physics* 148, 241728 (2018), 2018.
- Atsuto Seko, Hiroyuki Hayashi, Hisashi Kashima, and Isao Tanaka. Matrix- and tensor-based recommender systems for the discovery of currently unknown inorganic compounds. *Phys. Rev. Materials* 2, 013805, 2018.
- E. Swann, B. Sun, D. M. Cleland, and A.S. Barnard. Representing molecular and materials data for unsupervised machine learning. *Molecular simulation*, 2018.
- P. Villars and K. Cenzual (eds.). *Pearson's Crystal Data Crystal Structure Database for Inorganic Compounds*. ASM International, 2017.
- X. Yang, J. Zhang, K. Yoshizoe, K. Terayama, and K. Tsuda. ChemTS: an efficient Python library for de novo molecular generation. *Communications in materials informatics*, 2017.
- Mehrdad Yazdani. RemixNet: Generative Adversarial Networks for Mixing Multiple Inputs. In *Semantic Computing (ICSC)*, 2017.
- Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks. In *Computer Vision (ICCV), 2017 IEEE International Conference on*, 2017.