Phase-Mapper: An AI Platform to Accelerate High Throughput Materials Discovery

Yexiang Xue
Department of Computer Science
Cornell University
yexiang@cs.cornell.edu

Junwen Bai
Zhiyuan College
Shanghai Jiao Tong University, China
bjw_sjtu@sjtu.edu.cn

Ronan Le Bras, Brendan Rappazzo
Department of Computer Science
Cornell University
{rl454, bhr54}@cornell.edu

Richard Bernstein, Johan Bjorck, Liane Longpre
Department of Computer Science
Cornell University
{rab38, njb225, lfl42}@cornell.edu

Santosh K. Suram
Joint Center for Artificial Photosynthesis
California Institute of Technology
sksuram@caltech.edu

Robert B. van Dover
Materials Science and Engineering
Cornell University
rbv2@cornell.edu

John Gregoire
Joint Center for Artificial Photosynthesis
California Institute of Technology
gregoire@caltech.edu

Carla P. Gomes
Department of Computer Science
Cornell University
gomes@cs.cornell.edu

Abstract

High-throughput materials discovery involves the rapid synthesis, measurement, and characterization of many different but structurally related materials. A central problem in materials discovery, the phase map identification problem, involves the determination of the crystal structure of materials from materials composition and structural characterization data. We present Phase-Mapper, a novel solution platform that allows humans to interact with both the data and products of AI algorithms, including the incorporation of human feedback to constrain or initialize solutions. Phase-Mapper is compatible with any spectral demixing algorithm, including our novel solver, AgileFD, which is based on convolutive non-negative matrix factorization. AgileFD allows materials scientists to rapidly interpret XRD patterns, and can incorporate constraints to capture the physics of the materials as well as human feedback. We compare three solver variants with previously proposed methods in a large-scale experiment involving 20 synthetic systems, demonstrating the efficacy of imposing physical constraints using AgileFD. Since the deployment of Phase-Mapper at the Department of Energy's Joint Center for Artificial Photosynthesis (JCAP), thousands of X-ray diffraction patterns have been processed and the results are yielding discovery of new materials for energy applications, as exemplified by the discovery of a new family of metal oxide solar light absorbers, among the previously unsolved Nb-Mn-V oxide system, which is provided here as an illustrative example. Phase-Mapper is also being deployed at the Stanford Synchrotron Radiation Lightsource (SSRL) to enable phase mapping on datasets in real time.

1 Introduction

The wonders of modern technology can largely be attributed to advances in materials science that enable innovations from semiconductors to renewable energy. High-throughput materials discovery comprises a suite of emerging methodologies to rapidly identify new materials, especially undiscovered materials that are critical for next-generation technologies (Green, Takeuchi, and Hattrick-Simpers 2013). Specifically, high-throughput materials discovery involves rapidly synthesizing 10^2-10^3 unique materials comprising a "library" and rapidly screening them for properties of interest.

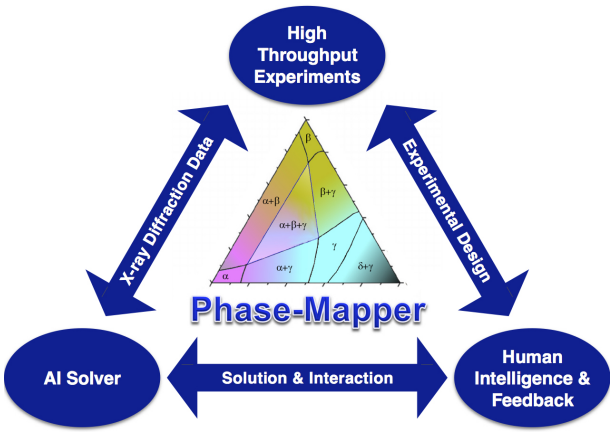

Figure 1: The Phase-Mapper platform integrates experimentation, AI solvers, and human feedback into a platform for high throughput materials discovery for discovering new materials.

To analyze the vast amount of data that are generated in high throughput experiments, automatic analysis becomes imperative. The traditional analysis workflow relies on iterative manual analysis and heuristics, resulting in months or years of analysis for a single library. This quickly becomes a bottleneck as humans are unable to keep up with the rate at which data are generated. The need for automatic and scalable tools provides unique opportunities to apply cutting edge techniques in computer science, AI, and data science to accelerate the materials discovery process.

In this paper we address the *phase mapping problem*,

a central problem in high-throughput materials discovery, which has critically lacked an efficient solution method. A material's phase describes a range of elemental composition and other conditions over which its properties and structure, the arrangement of the constituent atoms, change little. X-ray diffraction (XRD) is a ubiquitous technique to characterize crystal phases, as it produces a signal containing a series of peaks that serve as a "fingerprint" of the underlying atomic arrangement or crystal structure. Using traditional methods, materials scientists can obtain and interpret 1-10 XRD measurements per day, and with the recent development of automated, synchrotron-based XRD experiments, the measurement throughput has been accelerated to 10^3-10^5 measurements per day (Gregoire et al. 2009; 2014). The creation of a phase mapping algorithm that generates phase diagrams from these data remains an unsolved problem in materials science despite a series of advancements over the past decade (Hattrick-simpers, Gregoire, and Kusne 2016). The most pertinent need is to generate a physically meaningful phase diagram for the materials in a given library, which relies on the spectral demixing of the 10^2-10^3 XRD patterns into a small set of basis patterns (typically less than 10).

To address this substantial challenge, we developed **Phase-Mapper**, a comprehensive platform that tightly integrates XRD experimentation, AI problem solving, and human intelligence for the phase mapping problem (See Figure 1). In this platform, **within minutes**, an AI solver provides approximate results for the phase mapping problem, which are examined and further refined by materials scientists interactively and in real time. In addition, the results of Phase-Mapper can be used to further inform future experimental designs. The demixing algorithm is a cornerstone of the Phase-Mapper platform. We have developed a **novel solver called AgileFD**, based on convolutive non-negative matrix factorization (cNMF), a method that has been applied to blind source separation of audio signals and speech recognition (Smaragdis 2004; Mørup and Schmidt 2006).

AgileFD **features lightweight iterative updates of candidate solutions**, allowing it to complete many update iterations within a given time. In addition, we have developed a suite of adaptations that enable functionalities beyond cNMF. The extensions for AgileFD described here include incorporation of constraints to encode both **human input**, that capitalizes on a researcher's knowledge of a particular dataset, and *a priori* knowledge of the problem related to the **underlying physics** of phase diagrams. This, as demonstrated below, can be critical in obtaining physically meaningful solutions. In developing the Phase-Mapper platform, careful attention has been given to delivering a rich suite of capabilities while maintaining solver convergence times within minutes, which enables researchers to interact with the solver to refine the solution.

We compare three variants of AgileFD with previously proposed solvers on **an experiment involving 20 synthetically generated systems**. Our results show that AgileFD outperforms previous solvers in terms of reconstruction error and model correctness, because the model represents peak shifting in an efficient way. In addition, light weight update rules allow AgileFD to converge more quickly than previous solvers and with various extensions the solutions tend to be more physically meaningful.

Application Use and Payoff Phase-Mapper has been deployed at the Department of Energy's Joint Center for Artificial Photosynthesis (JCAP), where it has been used to run hundreds of phase mapping solutions on thousands of XRD measurements in the JCAP materials discovery pipeline.

We first encountered the phase mapping problem six years ago as part of our Computational Sustainability (Gomes 2009) effort to address pressing problems in renewable energy. Phase-Mapper is the culmination of our work since then in close collaboration with top experts in materials discovery. Over the course of this collaboration, we have made important contributions to the formal characterization of this problem, developed several synthetic instance generators, and developed several algorithms with theoretical and practical guarantees. We have also continuously developed tools to share experimental instance data, results, and solution visualizations with our collaborators throughout (Le Bras et al. 2011; Ermon et al. 2012; Le Bras et al. 2014; Ermon et al. 2015; Xue et al. 2015). Phase-Mapper is our most successful tool to date in this area; it removes many of the practical barriers to the use of previous methods, including better scalability, runtimes suitable for interactive use, and ease of access.

Prior to Phase-Mapper, the difficulty of interpreting X-ray diffraction data limited JCAP scientists' ability to take full advantage of resources to conduct high throughput experiments. Since the deployment of Phase-Mapper, **thousands of X-ray diffraction patterns have been processed and the results are yielding discovery of new materials for energy applications**. These are exemplified by **the discovery of a new family of metal oxide light absorbers in the previously unsolved Nb-Mn-V oxide system**, which is provided here as a case study and an illustrative example of the importance of encoding physical constraints to obtain physically meaningful phase diagram solutions. In light of the demonstrated computational efficiency and solution quality, Phase-Mapper is also being **deployed at the Stanford Synchrotron Radiation Lightsource (SSRL) to enable phase mapping on datasets in real time**.

We believe Phase-Mapper will lead to further developments in high-throughput materials discovery by providing rapid and critical insights into the phase behavior of new materials.

2 Phase-Mapper: AI for Materials Discovery

High throughput materials discovery is an experimentation pipeline for rapidly synthesizing, characterizing, and identifying new materials. In this pipeline, a handful of elements are deposited together on a two-dimensional substrate, so that different locations on the substrate receive varying proportions of the elements. This smooth variation in elemental composition across the substrate gives rise to the forming of a discrete set of materials, each of which is present on particular regions of the substrate. After deposition, sample locations on the substrate are probed with high energy X-rays, and because XRD patterns are indicative of crystal

structure, it is possible to characterize the discrete set of materials present.

A key challenge in solving the *phase-mapping problem* results from the fact that the XRD patterns obtained in the high throughput pipeline can be composed of a mixture of the XRD patterns of several materials. Therefore, *phase-mapping* is the problem of identifying the *characteristic* XRD patterns of the materials (or basis patterns or crystal structures of the materials) that demix the signal, and it lies at the heart of the analysis of high throughput data.

Mathematically, the measured XRD pattern in the j-th sample point can be characterized by a one dimensional signal $A_j(q)$. The "scattering vector magnitude" q is a monotonic transformation of the diffraction angle, and is directly related to the spacing of atoms in a crystal. The phase-mapping problem is to find a small number of phases $W_1(q), \ldots, W_K(q)$, and activation coefficients h_{ij}, such that the XRD patterns at each sample point can be explained by a linear combination of phases:

$$A_j(q) \approx \sum_{i=1}^{K} h_{ij} W_i(\lambda_{ij} q). \qquad \forall j \tag{1}$$

In the above definition, we write $W_i(\lambda_{ij}q)$ to allow for the phases to scale slightly according to parameter λ_{ij} at each sample point. This is the result of a commonly observed form of alloying, a process that can typically be approximated by a multiplicative scaling of the XRD pattern of a specific phase in the q domain. We also call this process "peak shifting", because the effect appears to make XRD patterns in the data shift to the left or to the right. In addition to the complications introduced by peak shifting, there are a number of other constraints on the solution of the phase-mapping problem, arising from the fact that the solution must describe a system constrained by the laws of physics. The most prominent is the so called Gibbs phase rule, which requires no more than three phases per sample point, in a ternary system, i.e., no more than three coefficients among h_{ij} for fixed j may be nonzero. Additionally, how the h_{ij} may vary spatially on the substrate, as well as the shapes that W_i may take, are constrained by physical laws.

Fundamentally novel techniques are required to solve the phase mapping problem quickly and accurately. A number of automatic techniques have been developed in recent years, which can be broadly grouped into clustering, constraint reasoning, and factor decomposition approaches. Proposed clustering methods such as hierarchical clustering (HCA) (Long et al. 2007), dynamic time warping kernel clustering (Le Bras et al. 2011), and mean shift theory (Kusne et al. 2014) produce maps of phase regions, but fail to resolve mixtures or identify basis patterns, and do not necessarily produce results consistent with physics. Constraint reasoning approaches, including satisfiability modulo theory (SMT) methods (Ermon et al. 2012), can provide physically meaningful results, but depend heavily on effective pre-processing, such as peak identification, and are computationally intensive. Approaches based on non-negative matrix factorization (NMF) (Long et al. 2009) are computationally efficient, but generally perform poorly when peak-shifting phenomena are present. CombiFD (Ermon et al. 2015) is another factor decomposition approach that uses combinatorial constraints to simultaneously enforce physical rules and accommodate peak shifting, but requires solving a combinatorial problem in each descent step, and is therefore computationally expensive.

Here we describe **Phase-Mapper**, an AI platform for rapidly solving the phase-mapping problem, integrating three key components: (i) cutting edge AI solvers; (ii) human intelligence and feedback; and (iii) high-throughput physical experiments. These components form an integrated process (see Figure 1):

- **Phase-Mapper** is supported by cutting edge AI solvers, including non-negative matrix factorization (Lee and Seung 2001) and CombiFD. We also highlight a new solver called AgileFD as a key component of the platform. Motivated by convolutive NMF, AgileFD features a set of light-weight updating rules and therefore a very fast gradient descent process. AgileFD is also flexible, allowing for the incorporation of additional physical constraints as well as human feedback through refinement.
- **Phase-Mapper** also provides tools for data exploration, visualization, and configuration that allows human experts as well as laypeople to analyze and improve solutions.
- **Phase-Mapper**'s solutions, obtained by the interaction between solvers and human users, can also shed light on the development of new physical experiments, for example by specifying regions of composition space to sample at higher resolution (active learning).

3 AgileFD: A Novel Phase-Mapping Solver

AgileFD is a key component in the Phase-Mapper platform. Compared with previously proposed methods for solving the phase-mapping problem, AgileFD features quick iterative updates of candidate solutions, which makes it possible for human experts to interact with the algorithm in real time. The key behind this speed lies in the efficient problem representation. Let the XRD patterns for all samples be represented by a matrix A where each column corresponds to one sample point and each row corresponds to $A_j(q)$ for a particular value of q. Under the assumptions of no noise and no shifting, i.e. for all i, j, $\lambda_{ij} = 1$, describing A as a linear combination of a few basis patterns $W_i(q)$ is equivalent to factorizing A as a product of two low rank matrices W and H. We enforce nonnegativity for W and H, which is required for the solutions to be physically meaningful.

$$A \approx W \cdot H = R.$$

Here, R denotes the approximate reconstruction of A. In this formulation, the columns of W form a set of basis patterns $W_i(q)$, while the columns of H corresponds to the values h_{ij} in equation 1. Previous approaches to solve the phase-mapping problem based on NMF have been unsuccessful in handling peak shifting, i.e. $\lambda_{ij} \neq 1$. The first contribution of AgileFD is to circumvent the shifting problem by a log space resampling. Under the variable substitution

$q \to \log q$ our signal becomes $W_i(\log q)$. More importantly, the shifted phase $W_i(\log \lambda q)$ becomes $W_i(\log \lambda + \log q)$, which transforms the multiplicative shift in the q domain into a constant additive offset. This allows the problem to be formulated in terms of convolutive nonnegative matrix factorization. After this variable substitution, we discretize the values of allowed λ and interpolate the signals at the corresponding geometric series q values. The problem can then be written:

$$A \approx \sum_m W^{\downarrow m} \cdot H^m = R. \tag{2}$$

Here, the columns of W still represent basis patterns. $W^{\downarrow m}$ is the result of shifting the rows of the W matrix down m rows, and padding the shifted m rows with 0, representing the basis patterns with a constant offset in the $\log q$ domain, which is equivalent to the original multiplicative shift in the q domain. The columns of H^m act as the activation of basis patterns for the basis patterns shifted down m units. Note that when $M = 1$, this formulation is equivalent to NMF aside from the log transformation.

AgileFD is a family of algorithms, which can be adapted to use different loss functions, regularization, and certain imposed constraints. Equation (2) is adapted from convolutive NMF (cNMF), which was first proposed to analyze audio signals (Smaragdis 2004). The phase-mapping problem differs from previous applications of cNMF for blind source separation as the $\log q$ domain is substituted for the time domain, and each source (phase) is expected to appear at most once per sample with a relatively small offset. As in cNMF, AgileFD uses a gradient descent approach to fit W and H. When the generalized Kullback-Leibler (KL) divergence is used in the objective function, gradient updates can be written multiplicatively, and are applied iteratively until convergence. See (Xue et al. 2016) for further details.

Lightweight Update Rules AgileFD's linear gradient update rules results in very fast convergence, typically within minutes. This is orders of magnitude faster than CombiFD, which uses a similar problem formulation but with combinatorial constraints explicitly enforced relying on a mixed integer programming solver. This increased efficiency of AgileFD enables high throughput analysis and also makes it possible for a human to interact with the system almost in real time.

Further Extensions of AgileFD for Materials Discovery Since the ultimate aim of the phase-mapping problem is to find a physically meaningful decomposition of the signal, for which the loss function is just a proxy, we must allow for experts to modify and inspect any candidate solution. It is not feasible to encode all physical constraints or the knowledge of a materials scientist within the solver *a priori*. Therefore, in the next few sections, we provide a number of novel modifications to the basic AgileFD algorithm, in order to impose prior knowledge or additional constraints derived from user interpretation of a proposed solution.

AgileFD with Frozen Values In the Phase-Mapper platform, the user is provided with the opportunity to freeze individual values in the W and H matrices. For example, a user might specify a known pattern or part of a previous solution as a basis pattern *a priori*, freezing the corresponding row or part thereof in W. Or the user might specify that a certain set of samples contain only a single material phase and set the non-corresponding H vales to zero. The result is an interactive, iterative matrix factorization.

Custom Initialization By initializing basis patterns or coefficients to values close to the expected solution, rather than random values, the user can direct the search to the correct solution space. We allow the user to specify basis patterns that can be taken from previous solutions, data samples, or provided manually, to use as an initial value. Similarly, initial values for the activation matrix can be specified.

Sparsity Regularization Sparse solutions are usually more easily interpreted, and in materials science they are more likely to be consistent with the underlying physics. The AgileFD system provides the option to introduce a soft penalty term for sparsity in H which can vary by index according to a human expert's preferences. Using L1-regularization for H and sparsity weight matrices γ_m, the sparse generalized KL-divergence objective function becomes:

$$\sum_{i,j} \left(A_{i,j} \log \frac{A_{i,j}}{R_{i,j}} - R_{i,j} + A_{i,j} \right) + \sum_m \|\gamma_m \circ H^m\|_1.$$

In order to avoid the degenerate solution where $H \to 0$, each basis pattern of W is L2-normalized at the beginning of each update iteration. In (Xue et al. 2016) we provide further details concerning the corresponding update rules for H.

The Gibbs Phase Rule In general, correct phase map solutions should follow the Gibbs phase rule, which specifies that the number of observed phases at a given chemical composition is no more than the number of chemical elements N_{el}:

$$\sum_i I_{ij} \leq N_{el}. \tag{3}$$

Here, I_{ij} is an indicator of whether phase i is present at sample location j. Materials scientists might also know *a priori*, or infer from previous proposed solutions, that certain regions contain fewer phases than the usual limit.

Such combinatorial constraints cannot be encoded directly in the update rules of AgileFD. One way to enforce these constraints, which has been used in previous methods such as CombiFD, is to directly encode it as hard combinatorial constraints. However, this results in a slow update process, as we have to solve a Mixed Integer Programming (MIP) problem in each iteration. As a novel routine, we apply the Gibbs phase rule by first solving the relaxed problem, then choose the best values to set to zero in H to satisfy Eq. 3, and then refine the solution by applying the update rules until convergence. Because the update rules are multiplicative, the zeroed values will remain zero.

Choosing the values to zero in H is independent for each sample point j. This can be solved greedily if a faster solution is desired, or using a MIP formulation, and/or successive rounds of constraints and refinement, if a more precise solution is desired with a somewhat longer wait time. This

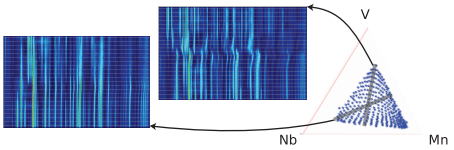

Figure 2: Two heatmaps of XRD patterns generated by taking slices in the visualizer.

extension is particularly useful when the unconstrained algorithm recovers a solution that is nearly correct except for relatively small violations of phase limits.

4 Phase Mapper: A Human-Machine Integrated Platform

We provide an integrated workflow with the Phase-Mapper platform, which includes visualizing and analyzing an instance file, setting the solver framework, analyzing the solution, and using that analysis to update the solver framework. The design objectives were simple: create a practical application that seamlessly connects a visualization system with a powerful solver that allows for interactive and large scale use. The main features of Phase-Mapper are the visualization tools and the solver interface.

Visualizer Phase-Mapper provides a way to visualize both the input data as well as the solution that is generated. When an instance file of a materials system is uploaded to the system, the visualizer will generate *a composition map*, which illustrates the varying compositions of elements, for all sample points. The user can freely inspect the XRD patterns of each sample point, as well as the heatmap of XRD patterns for a slice of sample points. A slice heatmap example is shown in Figure 2, where two selected *slices* of sample points are shown in grey. The heatmaps on the left represent the XRD patterns at the sample points in the slice.

When the solution files are loaded into the application, either uploaded by the user or generated by the solver, three new plots are generated: 1) the basis patterns that were found as solutions; 2) a composition map displaying the mixture proportions; and 3) the original XRD signal compared with the reconstructed signal for a user specified sample point.

Connection to Solver The solving feature of Phase-Mapper enables users to interact with the AI solver behind the scenes. The user can specify many solver parameters such as how much to enforce sparsity, how many phases the solution should have, and how much shift between basis patterns it should allow. The user can also specify initial or frozen values to use as basis patterns.

User input parameters aim to help the solver more efficiently and accurately find a solution, either by starting the solver off closer to a solution, or distorting the solution space so the solver finds a more accurate solution.

5 System Design and Deployment

Application Description Phase-Mapper was originally developed as a standalone application in C#, built on our previous visualization tool, UDiscoverItViz (Le Bras et al. 2014). As this restricted users to the Windows platform and interactive-only use, we reimplemented the AgileFD solver in C++ as a commandline tool, levaraging Armadillo as a structured BLAS interface for numerical computations, and CPLEX for the application of combinatorial constraints. This implementation is flexible and can be run on a personal computer, in batch on an HPC cluster, or connected to a graphical interface.

In order to support all use cases and also for platform independence, we rewrote the visualization component for the web. The visualization functions were implemented entirely client-side using HTML5 canvas with the help of the W3.CSS framework and JQuery. The user can load and explore locally stored or solver-generated instance and solution files, with a variety of visualization features.

AgileFD is also integrated with Phase-Mapper as a stateless web service. In order to invoke the solver, the user loads the instance file, and selects configuration parameters through the GUI. The client then makes an AJAX request to the web service, which is a PHP script that runs AgileFD on the server using a system call. On completion, the solution file data is returned to the client and loaded into the visualization interface, and can be saved locally if desired. Because the requests are asynchronous, the user can continue to explore the data or previous solutions while AgileFD runs on the server.

Development and Deployment Development of each of the main components was led by a separate developer who implemented the majority of their component. However, major design, architecture, and implementation decisions were discussed collectively. The developers are a mix of graduate and undergraduate students, and professional staff. We have weekly webconferences between the computer science group at Cornell University and the materials science group at JCAP, in which materials scientists provide feedback on both the solver and the interface and suggest modifications and new features that would be useful to them. Each component required about a month of development to reach a stable version, with continuous development after the initial release.

Maintainance The primary maintenance activity is continuous development of both AgileFD and the Phase-Mapper interface, which must be kept in sync. These updates are made as they are ready. The system is stateless, which simplifies maintenance: modifications to AgileFD can be deployed as a drop-in replacement; similarly, Phase-Mapper can be updated like any website. Additionally, supporting increasing demand can be accomplished using elastic scaling through a cloud-based hosting service.

6 Experiments

6.1 Large Scale Experiments

Despite the fact that several solvers have been proposed to solve the phase-mapping problem in recent years, most solvers were configured and tested only on a handful of systems. Here we provide a large-scale evaluation of various solvers on the phase-mapping problem.

We generated synthetic ternary metallic systems using data provided by the Materials Project (Jain et al. 2013),

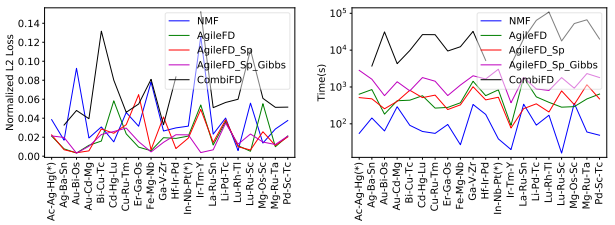

Figure 3: (Left) Normalized L2 loss vs. ground truth phases for NMF, AgileFD, AgileFD with sparsity, AgileFD with sparsity & the Gibbs phase rule and CombiFD for 20 physical systems. AgileFD and its variants perform best. (Right) Runtimes to solve 20 instances. CombiFD was run on 12 cores with a soft time limit of 1 hour or until it found a feasible solution (1-2 iterations; convergence would take days); (*) indicates that CombiFD did not complete within 36 hours. The times for the other solvers are until convergence using 1 core (convergence gap 2×10^{-5}).

which provides theoretical crystal structure information using Density-Functional Theory based convex hull construction in composition-energy space that predicts compositions and corresponding lowest energy of formation atomic configurations that form the vertices of the convex hull.

The XRD patterns at compositions between the vertices of the convex hull were calculated, using pymatgen (Ong et al. 2013), by interpolation of a) XRD patterns of the vertex phases or b) modified XRD patterns of the vertex phases to accommodate alloying. We applied a stylized model of solid solubility and alloying, and used structure interpolation to simulate modified phase diagrams that include the additional degrees of freedom from alloying. We calculated XRD patterns for each modified constituent, including their interpolated structures, and combined them according to the mixture proportions in the phase diagram.

We selected 20 examples containing varying numbers of phases and amounts of alloying for our experminent, reflecting many properties of real experimental data. The simulation used to generate these data provides ground truth, which we compare directly to computed solutions found by different solvers. We tested NMF (implemented as AgileFD with $M = 1$), AgileFD, AgileFD with sparsity regularization (AgileFD-Sp), AgileFD with sparsity and the Gibbs phase rule enforced (AgileFD-Sp-Gibbs), and CombiFD. The convergence gap for NMF and AgileFD are 2×10^{-5}, and the sparsity regularization parameter is 0.35 in AgileFD Sp. CombiFD uses a MIP gap of 0.2, and a soft time limit of 1 hour on 12 cores, which is exceeded to reach a feasible solution and typically allows only 1 update iteration. This is the reason for reduced solution quality. The other solvers use 1 core and run until convergence. We assume K and M are provided. A method for automatically selecting K and M is important, but is beyond the scope of this study.

First we evaluate the solution quality by comparing the modeled signal for each phase at each sample point, including shift, to the known signal from that phase at that sample point. We find the permutation of the phases in the solution to best match the ground truth, and calculate the L2 loss for each component. These are summed over all phases and samples, and scaled by the total value of all signals.

System	K	NMF	AgileFD	AgileFD Sp	AgileFD Sp Gibbs	CombiFD
Ac-Ag-Hg (*)	5	0.35	0.28	0.43	**1.00**	*
Ag-Ba-Sn	13	0.09	0.09	0.21	**1.00**	**1.00**
Au-Bi-Os	4	0.81	0.77	0.81	**1.00**	**1.00**
Au-Cd-Mg	12	0.12	0.12	0.20	**1.00**	**1.00**
Bi-Cu-Tc	3	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**
Cd-Hg-Lu	7	0.12	0.11	0.22	**1.00**	**1.00**
Cu-Ru-Tm	6	0.32	0.27	0.62	**1.00**	**1.00**
Er-Ga-Os	8	0.08	0.12	0.43	**1.00**	**1.00**
Fe-Mg-Nb	4	0.73	0.78	0.89	**1.00**	**1.00**
Ga-V-Zr	13	0.10	0.07	0.41	**1.00**	**1.00**
Hf-Ir-Pd	8	0.27	0.38	0.51	**1.00**	**1.00**
In-Nb-Pt (*)	10	0.14	0.14	0.34	**1.00**	*
Ir-Tm-Y	4	0.86	0.94	0.98	**1.00**	**1.00**
La-Ru-Sn	12	0.12	0.09	0.32	**1.00**	**1.00**
Li-Pd-Tc	8	0.25	0.48	0.49	**1.00**	**1.00**
Lu-Rh-Tl	9	0.17	0.12	0.36	**1.00**	**1.00**
Lu-Ru-Sc	4	0.51	0.95	0.95	**1.00**	**1.00**
Mg-Os-Sc	7	0.38	0.35	0.57	**1.00**	**1.00**
Mg-Ru-Ta	8	0.22	0.20	0.27	**1.00**	**1.00**
Pd-Sc-Tc	9	0.24	0.14	0.44	**1.00**	**1.00**

Table 1: Percentage of sample points that satisfy the Gibbs phase rule for each solver on 20 physical systems. Phases that account for less than 1% of the modeled signal are not counted towards this phase limit, and (*) indicates that CombiFD did not complete.

As shown in Figure 3 (Left), in general the solutions found by AgileFD (including AgileFD-Sp and AgileFD-Sp-gibbs) better match the ground truth when compared with NMF and CombiFD. NMF underperforms because it cannot model peak shifting. Notice that CombiFD, along with other solvers, is given a short time limit (one hour), which is different from the experimental setting in (Ermon et al. 2015). This short time limit is more in line with the way materials scientists use Phase Mapper. Under this short time limit, CombiFD often only completed a few iterations, and in a few cases timed out in the first iteration, in contrast with AgileFD and NMF, which completed thousands of updates. The expensive updates of CombiFD resulted in lower solution quality. Figure 3 (Right) compares the runtimes on each instance. The full table of the number of updates is included in the technical report (Xue et al. 2016).

AgileFD-Sp and AgileFD-Sp-Gibbs outperform AgileFD without extensions. They are able to find solutions that better match the physical constraints. We calculate the percentage of sample points that satisfy the Gibbs phase rule, for the solutions found by each solver for each instance. The result is shown in Table 1.

6.2 Case Study: Nb-V-Mn Oxides

The integration of the rapid solver with visualization tools enables materials scientists to take advantage of the tunable initialization and constraint parameters to create a meaningful solution. An illustrative example is found in a ternary composition library containing a broad range of compositions in the Nb-V-Mn oxide space. While the phase behavior

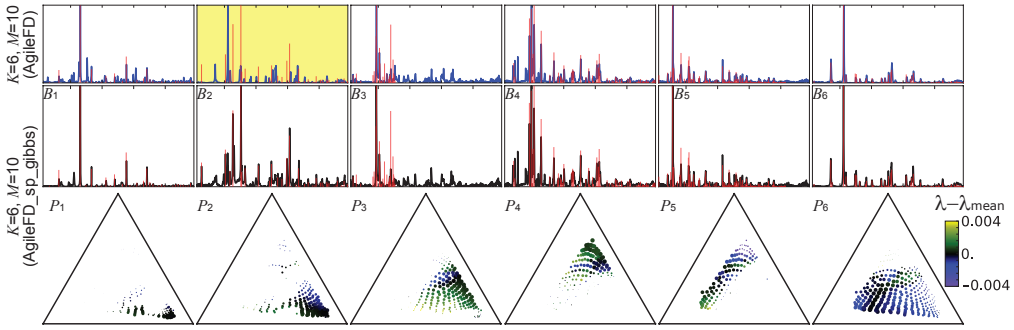

Figure 4: The K=6 basis patterns for the AgileFD solution (top) and AgileFD-Sp-Gibbs solution (middle) for the Nb-V-Mn system are shown along with stick patterns (translucent red) of the crystal structures identified using the AgileFD-Sp-Gibbs solution. The primary discrepancy in the AgileFD solution is highlighted in yellow. The phase map representation of H is shown as a series of composition plots for each phase from the AgileFD-Sp-Gibbs solution (bottom) showing the samples containing each phase with point size indicating the concentration of the phase and the point color indicating the fractional shift with respect to the plotted basis patterns. The elemental labels for the composition diagrams are shown in Figure 5.

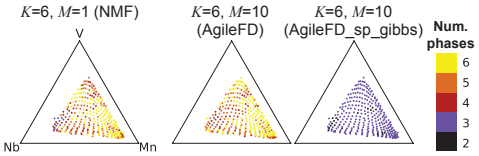

Figure 5: Composition maps of the number of the K=6 basis components utilized in the solutions from 3 different solvers: NMF, AgileFD, and AgileFD-Sp-Gibbs. Values in excess of 3 are non-physical.

of binary sub-compositions (e.g. Nb-V oxides) have been previously studied, the ternary compositions are being explored for the first time to discover solar light absorbers for energy applications. While the set of 317 XRD patterns provides sufficient information to solve the phase behavior of these oxides, the materials researchers were unable to obtain a meaningful phase diagram for over a year due to the complex phase behavior that evaded comprehension via manual analysis. Using the visualization tools and rapidly trying various solutions, the researchers determined that there are K=6 primary phases in this dataset. For this case study, we compare solutions from 3 different solvers with K=6: (1) NMF; (2) AgileFD with M=10, which includes shifting of each basis pattern by up to approximately 2%; (3) AgileFD-Sp-Gibbs with M=10.

Adherence to the Gibbs phase rule is important for providing the researcher with a meaningful phase diagram. The number of basis patterns utilized for each sample is shown in Figure 5 for each of the 3 solutions. The NMF solution adheres to the Gibbs phase rule for only 4% of the samples, with nearly half of the samples utilizing all 6 basis patterns. While the AgileFD solution is more meaningful than the NMF solution due to the tracking of alloying via basis pattern shifting, the AgileFD solution uses more phases for several compositions, corresponding to a worse violation of the Gibbs phase rule. This property of the AgileFD solver may be understood intuitively by considering that for a given sample, any of the M copies of each basis pattern can be utilized to model small features in the XRD patterns, resulting in small amounts of shifted patterns to appear across the composition space and further motivating the encoding of the Gibbs phase rule in the solver. With this constraint, the AgileFD solution utilizes a maximum of 3 phases and in some composition regions only 2 phases are utilized.

Figure 4 summarizes the AgileFD-Sp-Gibbs solution, and its attributes. Even though some of the solution characteristics were not explicitly enforced, they are desirable, since they provide meaning to the researcher, namely: (1) excellent matching of the primary peaks of each basis pattern with a known crystal structure, demonstrating that each basis pattern is truly representative of a phase, (2) excellent composition space connectivity of each phase concentration map, as expected for equilibrium phase behavior, (3) systematic compositional variation in the shift parameter λ, demonstrating alloying within the phases, in particular phases 3, 5, and 6. While the AgileFD solution includes pattern shifting, the lack of adherence to the Gibbs phase rule has important consequences on the solution that cannot be ameliorated through modification of the H matrix to impose Gibbs phase rule *ex post facto*, without a corresponding refinement of W. The most prominent difference, in comparison to the AgileFD-Sp-Gibbs solution, is highlighted in Figure 4 where the second basis pattern of the AgileFD basis pattern is quite different from that of the AgileFD-Sp-Gibbs solution. This AgileFD basis pattern contains a mixture of signals from other phases and is thus not able to be matched to a known structure. Enforcement of the Gibbs phase rule in the solver results in effective demixing of basis patterns such that all 6 primary phases are identified. Indeed, only by imposing *a priori* constraints in AgileFD is a complete, meaningful solution produced.

7 Conclusion

High-throughput materials discovery is revolutionizing the efficiency of materials science. A major, critically-missing component of the high-throughput materials discovery

pipeline is the ability to rapidly solve the phase map identification problem, which involves the determination of the underlying phase diagram of a family of materials from their composition and structural characterization data. To address this challenge, we developed Phase-Mapper, a comprehensive platform that tightly integrates XRD experimentation, AI problem solving, and human intelligence. AI solvers in Phase-Mapper provide high-quality solutions to the phase mapping problem within minutes, which can then be examined and further refined by materials scientists interactively and in real time. We have developed a novel solver, AgileFD, that features lightweight iterative updates of candidate solutions and a suite of adaptations to the multiplicative update rules. In particular, we have developed the ability to incorporate constraints that capture the physics of materials as well as human feedback, enabling functionalities well beyond traditional demixing techniques and producing physically-meaningful solutions. We compare different solver variants with previously proposed methods in a large-scale experiment, demonstrating the efficacy of imposing physical constrains using AgileFD, while keeping fast solution times. Phase-Mapper has been deployed at the Department of Energy's Joint Center for Artificial Photosynthesis (JCAP) for materials scientists to solve a wide variety of real-world phase diagrams. Since the deployment of Phase-Mapper, thousands of X-ray diffraction patterns have been processed and the results are yielding the discovery of new materials for energy applications, as exemplified by the discovery of a new family of metal oxide solar light absorbers, among the previously unsolved Nb-Mn-V oxide system, which is provided here as an illustrative example. Phase-Mapper is also being deployed at the Stanford Synchrotron Radiation Lightsource (SSRL) to enable phase mapping on datasets in real time. We believe Phase-Mapper will lead to further developments in high-throughput materials discovery by providing rapid and critical insights into the phase behavior of new materials.

Acknowledgements This material is supported by NSF awards CCF-1522054 and CNS-0832782 (Expeditions), CNS-1059284 (Infrastructure), and IIS-1344201 (INSPIRE); and ARO award W911-NF-14-1-0498. Materials experiments are supported through the Office of Science of the U.S. Department of Energy under Award No. DE-SC0004993. Use of the Stanford Synchrotron Radiation Lightsource, SLAC National Accelerator Laboratory, is supported by the U.S. Department of Energy, Office of Science, Office of Basic Energy Sciences under Contract No. DE-AC02-76SF00515.

References

Ermon, S.; Le Bras, R.; Gomes, C. P.; Selman, B.; and van Dover, R. B. 2012. Smt-aided combinatorial materials discovery. In *SAT*.

Ermon, S.; Le Bras, R.; Santosh, S.; Gregoire, J. M.; Suram, S. K.; Gomes, C. P.; Selman, B.; and van Dover, R. B. 2015. Pattern decomposition with complex combinatorial constraints: Application to materials discovery. In *AAAI*.

Gomes, C. P. 2009. Computational sustainability: Computational methods for a sustainable environment, economy, and society. *The Bridge* 39(4):5–13.

Green, M. L.; Takeuchi, I.; and Hattrick-Simpers, J. R. 2013. Applications of high throughput (combinatorial) methodologies to electronic, magnetic, optical, and energy-related materials. *Journal of Applied Physics* 113.

Gregoire, J. M.; Dale, D.; Kazimirov, A.; DiSalvo, F. J.; and van Dover, R. B. 2009. High energy x-ray diffraction/x-ray fluorescence spectroscopy for high-throughput analysis of composition spread thin films. *The Review of Scientific Instruments* 80(12).

Gregoire, J. M.; Van Campen, D. G.; Miller, C. E.; Jones, R. J. R.; Suram, S. K.; and Mehta, A. 2014. High-throughput synchrotron X-ray diffraction for combinatorial phase mapping. *Journal of synchrotron radiation* 21.

Hattrick-simpers, J. R.; Gregoire, J. M.; and Kusne, A. G. 2016. Perspective : Composition structure property mapping in high-throughput experiments : Turning data into knowledge. *APL Materials* 4.

Jain, A.; Ong, S. P.; Hautier, G.; Chen, W.; Richards, W. D.; Dacek, S.; Cholia, S.; Gunter, D.; Skinner, D.; Ceder, G.; and Persson, K. A. 2013. The Materials Project: A materials genome approach to accelerating materials innovation. *APL Materials* 1.

Kusne, A.; Gao, T.; Mehta, A.; Ke, L.; Cuong Nguyen, M.; Ho, K.-M.; Antropov, V.; Wang, C.-Z.; Kramer, M. J.; Long, C.; and Takeuchi, I. 2014. On-the-fly machine-learning for high-throughput experiments: search for rare-earth-free permanent magnets. *Scientific Reports 4* (6367).

Le Bras, R.; Damoulas, T.; Gregoire, J.; Sabharwal, A.; Gomes, C.; and van Dover, R. 2011. Constraint reasoning and kernel clust. for pattern decomp. with scaling. In *CP*.

Le Bras, R.; Bernstein, R.; Gregoire, J. M.; Suram, S. K.; Gomes, C. P.; Selman, B.; and van Dover, R. B. 2014. A computational challenge problem in materials discovery: Synthetic problem generator and real-world datasets. In *Proceedings of the 28th International Conference on Artificial Intelligence*, AAAI'14.

Lee, D. D., and Seung, H. S. 2001. Algorithms for non-negative matrix factorization. In *NIPS*, 556–562.

Long, C.; Hattrick-Simpers, J.; Murakami, M.; Srivastava, R.; Takeuchi, I.; Karen, V.; and Li, X. 2007. Rapid structural mapping of ternary metallic alloy systems using the combinat. approach and cluster analysis. *Rev. Sci. Inst.* 78.

Long, C.; Bunker, D.; Karen, V.; Li, X.; and Takeuchi, I. 2009. Rapid identification of structural phases in combinatorial thin-film libraries using x-ray diffraction and non-negative matrix factorization. *Rev. Sci. Instruments* 80.

Mørup, M., and Schmidt, M. N. 2006. Sparse non-negative matrix factor 2-d deconvolution. Technical report.

Ong, S. P.; Richards, W. D.; Jain, A.; Hautier, G.; Kocher, M.; Cholia, S.; Gunter, D.; Chevrier, V. L.; Persson, K. A.; and Ceder, G. 2013. Python Materials Genomics (pymatgen): A robust, open-source python library for materials analysis. *Computational Materials Science* 68.

Smaragdis, P. 2004. Non-negative matrix factor deconvolution; extraction of multiple sound sources from monophonic inputs. In *Independent Component Analysis and Blind Signal Separation*.

Xue, Y.; Ermon, S.; Gomes, C. P.; and Selman, B. 2015. Uncovering Hidden Structure through Parallel Problem Decomposition for the Set Basis Problem: Application to Materials Discovery. *IJCAI* 146–154.

Xue, Y.; Bai, J.; Le Bras, R.; Rappazzo, B.; Bernstein, R.; Bjorck, J.; Suram, S. K.; van Dover, R. B.; Gregoire, J.; and Gomes, C. P. 2016. Phase Mapper: an AI Platform to Accelerate High Throughput Materials Discovery. *Technical Report*.