Third-dimension information retrieval from a single convergent-beam transmission electron diffraction pattern using an artificial neural network

Robert S. Pennington,* Wouter Van den Broek, and Christoph T. Koch
Institute for Experimental Physics, Universität Ulm, Albert-Einstein-Allee 11, 89081 Ulm, Germany
(Received 28 November 2013; revised manuscript received 5 March 2014; published 8 May 2014)

We have reconstructed third-dimension specimen information from convergent-beam electron diffraction (CBED) patterns simulated using the stacked-Bloch-wave method. By reformulating the stacked-Bloch-wave formalism as an artificial neural network and optimizing with resilient back propagation, we demonstrate specimen orientation reconstructions with depth resolutions down to 5 nm. To show our algorithm's ability to analyze realistic data, we also discuss and demonstrate our algorithm reconstructing from noisy data and using a limited number of CBED disks. Applicability of this reconstruction algorithm to other specimen parameters is discussed.

DOI: 10.1103/PhysRevB.89.205409 PACS number(s): 61.05.J−, 07.05.Mh, 68.37.Lp, 68.65.Ac

I. INTRODUCTION

The transmission electron microscope (TEM) uses a relativistic electron beam for materials characterization with local specificity ranging from micrometers to sub-Angstrom, and can determine chemical composition, crystallographic structure, electric and magnetic potentials, and morphology [1–3]. However, the beam electrons pass through the entire specimen before reaching the detector, so they yield two-dimensional "flat" information about a three-dimensional object. Due to the strong forward scattering of electrons at these energies (e.g., 80 keV), specimen properties, including crystal orientation, thickness, strain state, electronic state, and chemical composition are projected along the "third dimension" (along the beam direction), but this projection is highly nonlinear due to multiple scattering from the strong beam-specimen interaction, so retrieving these properties from TEM data is a difficult inverse problem [2,4–6]. Existing techniques for retrieving third-dimension information, including electron tilt tomography or scanning confocal electron microscopy [7–12], typically only retrieve specimen density. Some third-dimension atomic-level local structural information is encoded in weak higher-order-Laue-zone reflections [2]; in this work, we only consider and simulate strong zero-order-Laue-zone reflections. To improve third-dimension TEM materials characterization, we have developed an iterative algorithm for retrieving any encoded specimen properties from a single elastically filtered TEM convergent-beam electron diffraction (CBED) pattern including multiple scattering. Our algorithm combines forward simulation with efficient optimization tools [13,14] from artificial-neural-network (ANN) theory, yielding iterative refinement of specimen parameters from TEM diffraction data. In this work, we test this algorithm on data with and without noise, as well as for CBED data with only a few measured disks.

Forward-simulation algorithms that include multiple scattering are a conventional tool to determine TEM data from known specimen and imaging conditions. These algorithms, such as multislice [15,16] and Bloch-wave [4,17–19], can simulate TEM data that can agree quantitatively with experimental data [20–22], including convergent-beam electron diffraction (CBED) [23,24], though discussion continues about other imaging modes [25–27]. These algorithms include multiple scattering, also called dynamical diffraction, which is necessary for all but the thinnest TEM specimens [6]. Alternatively, electron scattering under the assumption that the electron scatters only once, known as kinematical scattering [19], is easier to compute, but only useful for thick specimens if the specimen is oriented to an extremely weak diffraction condition. Even then, as noted below, the recorded data may still be influenced by dynamical diffraction, making kinematical simulations unreliable. While kinematical simulations can be directly inverted to determine third-dimensional properties [28], they may not present an accurate picture of the specimen [29]. Consequently, accurate forward-simulation algorithms are computationally intensive [30], so brute-force multiparameter specimen optimization is impractical. Our algorithm is able to efficiently perform third-dimension multiparameter optimization using techniques derived from reformulating the stacked-Bloch-wave (SBW) [30–33] forward-simulation algorithm as an ANN [13,14], similar to treatments of the atomic-resolution multislice algorithm [34,35].

Our algorithm is applicable to optimizing any of the third-dimension material properties detailed above, in principle, from crystal orientation to chemical composition. Here, we focus on only local crystal orientation, which can help explain asymmetric features of convergent-beam electron diffraction (CBED) patterns. A related parameter, strain state, has been characterized in projection with multiple experimental TEM methods in, e.g., semiconductor devices [36]. To determine third-dimension strain state information, strain-model-based analysis and CBED pattern analysis of the strain state in projection have been reported [37–39]. Unlike our method, these require accurate prior knowledge of the constrained domain of possible specimen strain states. While single-scattering-only simulations were previously used to investigate strain in ion-implanted surface layers from rocking curves in CBED patterns [28], those pattern features were later shown to be sensitive to both dynamical diffraction and a limited strain range [29], which demonstrates that neglecting multiple scattering can lead to incomplete data interpretation. Figure 1 shows a schematic of the CBED geometry, including the electron beam and a multilayer specimen, and a CBED pattern produced by forward-simulation discussed in Secs. II and III from a specimen with no local crystal orientation

*Corresponding author: robert.pennington@uni-ulm.de

 ©2014 American Physical Society

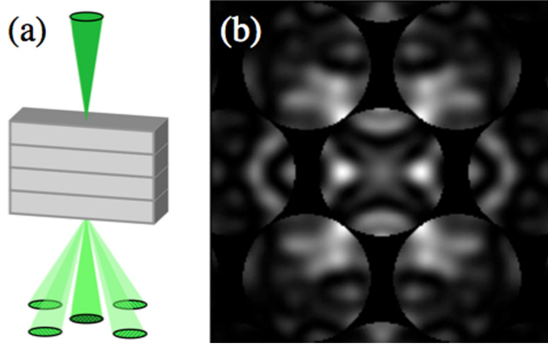

FIG. 1. (Color online) CBED imaging process and simulated CBED pattern. (a) Schematic of the CBED imaging process [2]. (b) ($2^\circ \times 2^\circ$) CBED pattern from 100 nm-thick unstrained Si [110], on-axis with respect to the beam, simulated at 80 kV with 197 eigenstates from the zero-order-Laue-zone, determined through excitation-error filtering of the on-axis orientation ($s_z = 1.225\,\text{nm}^{-1}$), and a 6.5 mrad condenser aperture, with 0.01° per simulated pixel. Literature values were used for scattering factors, Debye-Waller factors, and lattice parameters [40–42]. Inelastic scattering used the Bird-King model [43].

variation along the third-dimension. By comparison, Fig. 2 shows two simulated CBED patterns. In Fig. 2(a), the left-right asymmetry arises from third-dimension layer-by-layer local crystal orientation variation. When tilting the specimen stage (i.e., a constant third-dimensional crystal orientation) to attempt to match the asymmetric pattern in Fig. 2(a), the best-fit pattern, in Fig. 2(b), does not reproduce the asymmetric features of the target pattern. Therefore local crystal orientation could be experimentally detectable and would require layer-by-layer simulations.

II. THEORY

For our layer-by-layer simulations, we use the stacked-Bloch-wave (SBW) forward-simulation algorithm, proposed

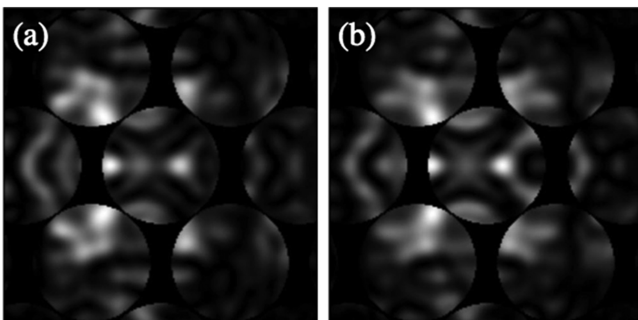

FIG. 2. Simulated asymmetric CBED pattern and best shifted-symmetric match. (a) Asymmetric CBED pattern under the same imaging conditions and reflections as Fig. 1, with layer-by-layer specimen tilts in the [001] direction relative to the zone axis of $\{a,a,b,c,c,d,d,c,b,a\}$, where $a = 0.00^\circ$, $b = -0.06^\circ$, $c = -0.15^\circ$, and $d = -0.20^\circ$. Each layer was 10-nm thick. (b) Best-fitting shifted-symmetric CBED pattern to the asymmetric left pattern, with specimen tilt of -0.1269° (rounded). The pattern appears shifted, but without the left pattern's asymmetry.

and used previously [30–33]. We reformulate SBW as an artificial neural network (ANN), which enables the backpropagation of error (BPOE) [13] and resilient backpropagation (Rprop) [14] methods, providing efficient layer-by-layer parameter optimization. This reformulation does not change the SBW algorithm itself, but does allow for efficient optimization tools to be applied. In this section, we discuss the SBW algorithm and the ANN-based optimization tools we apply to it.

An SBW simulation, like the Bloch-wave method, decomposes the electron beam into s eigenstates comprising ψ, a length-s complex vector, with each eigenstate corresponding to an allowed crystallographic direction in the specimen. The beam-specimen scattering matrix is given by the complex matrix $\mathsf{S}(P) = \exp[\imath t \lambda \mathsf{A}(P)]$ of dimensions $s \times s$, where $\mathsf{A}(P)$ is a specimen property matrix, t is the specimen thickness, and λ is the electron wavelength [44]. $\mathsf{A}(P)$ depends on a set P of specimen parameters. P can include atomic scattering and structure factors, chemical composition, beam absorption, lattice parameter, atomic thermal displacement (Debye-Waller factor), beam tilt, and local crystal orientation relative to the electron beam, as discussed above. Due to the matrix exponential used to generate $\mathsf{S}(P)$ from $\mathsf{A}(P)$, changing e.g., only the diagonal elements of A changes all the elements of the corresponding S.

A SBW simulation decomposes the specimen into L layers with parameters $P_{\text{all}} = \{P_1, P_2, \ldots, P_L\}$, from which are generated the set of matrices $\mathsf{S}_{\text{all}} = \{\mathsf{S}_1, \mathsf{S}_2, \ldots, \mathsf{S}_L\}$. The incident beam ψ_0 is forward-propagated through the specimen by successive matrix-vector multiplications:

$$\begin{aligned}
\psi_1 &= \mathsf{S}_1 \psi_0 = \mathsf{S}(P_1)\psi_0, \\
\psi_2 &= \mathsf{S}_2 \psi_1 = \mathsf{S}_2 \mathsf{S}_1 \psi_0 = \mathsf{S}(P_2)\mathsf{S}(P_1)\psi_0, \\
&\vdots \\
\psi_L &= \mathsf{S}_L \psi_{L-1} = \mathsf{S}_L \mathsf{S}_{L-1} \cdots \mathsf{S}_1 \psi_0 \\
&= \mathsf{S}(P_L)\mathsf{S}(P_{L-1}) \cdots \mathsf{S}(P_1)\psi_0.
\end{aligned}$$

We define the set $\psi_{\text{all}} = \{\psi_0, \psi_1, \psi_2, \ldots, \psi_{L-1}, \psi_L\}$ containing the beam state at the specimen entrance surface ψ_0, at the specimen exit surface ψ_L, and between each layer in the material. The final intensity at the detector is $I = |\psi_L|^2$.

The SBW algorithm corresponds to a feed-forward simply-connected ANN [13] with the set of weights S_{all} and the set of nodes ψ_{all}. Because S_{all} is a function of P_{all}, optimizing the weights yields optimized specimen parameters. To efficiently optimize over many specimen parameters, including multiple specimen parameters in the same layer, we use the BPOE and Rprop ANN iterative algorithms as follows.

In optimization-iteration m, a candidate set $P_{\text{all},m}$ is used to generate a set of $\mathsf{S}_{\text{all},m} = \mathsf{S}(P_{\text{all},m})$. After forward propagation, the candidate exit intensity for beam eigenstate (diffraction reflection) k, I_k, is compared with the measured exit intensity for the same eigenstate, J_k, yielding the individual-reflection mismatch error $E_k = \frac{1}{2}|I_k - J_k|^2$, which are summed to determine the mismatch error $E = \sum_k E_k$ for that beam tilt. Applying BPOE yields the full set of $\partial E / \partial \mathsf{S}_{\text{all},m}$ at the cost of only a single reverse pass through the ANN.

We now discuss the implementation of the BPOE method for the SBW algorithm. The BPOE method for determining the

derivatives of the error function E requires working from the output layer and moving "back" through the ANN (in this case, through the SBW algorithm). The BPOE method has been previously applied to the multislice algorithm [35], but because the multislice and SBW forward-propagation algorithms are different, the BPOE method also operates differently for multislice and SBW. First, the error derivative for a parameter p that affects one layer l, $\partial E/\partial p$, is

$$\frac{\partial E}{\partial p} = \sum_k 2 \cdot \mathrm{Re}\left[U_{l,k} \cdot \left(\frac{\partial \mathsf{S}_l}{\partial p} \times \psi_{l-1}\right)_k\right], \quad (1)$$

where K is the set of simulated reflections and the length of K is $|K| = s$, $k \in K$ is a single reflection, S_l is the SBW scattering matrix for layer l, and U is the ANN prefactor (an s-length vector). Twice the real part is taken because of the complex-valued nature of the forward-simulation algorithm [35].

The kth element of the ANN prefactor vector U is, at the exit surface after all L layers:

$$U_{L,k} = (I_k - J_k)\psi^*_{L,k}, \quad (2)$$

where $\psi_{L,k}$ is the kth reflection of the Lth (and final) layer, and I_k and J_k are, respectively, the simulated and measured beam intensities of reflection k. For the other layers, U for the $(l-1)$th layer is generated from U for the lth layer:

$$U_{l-1} = \mathsf{S}_l{}^T \times U_l. \quad (3)$$

The BPOE derivatives $\partial E/\partial p$ are calculated using three equations, (1)–(3), but require $\partial \mathsf{S}/\partial p$, as discussed in the next paragraph.

Using BPOE derivatives and Rprop, we can iteratively update P_{all}. Because each parameter in P_{all} only affects one member of $\mathsf{S}_{\text{all},m}$, the partial derivative of the intensity mismatch E with respect to a single parameter $p \in P_{\text{all}}$ in layer l is determined by combining the BPOE weight derivatives for layer l and $\partial \mathsf{S}/\partial p$. We chose to evaluate $\partial \mathsf{S}/\partial p$ using finite-difference derivatives, which provides maximum flexibility in parameter choice. Rprop is then applied to each individual $p_m \in P_{\text{all},m}$ to determine $P_{\text{all},m+1}$ using the signs of the partial derivatives for each parameter from the mth and $(m-1)$th iterations and the previous update value $\Delta p_m = p_m - p_{m-1}$; if the partial derivative sign changes, then $p_{m+1} = p_m - 0.5\Delta p_m$, else $p_{m+1} = p_m + 1.2\Delta p_m$ [14]. If some elements of P_{all} have known true values, this optimization algorithm can be applied to the subset of unknown values $Q_{\text{all}} \subseteq P_{\text{all}}$.

CBED patterns require sequential SBW simulations. Assuming the entire pattern uses the same set of eigenstates s, electrons along one incident beam vector are scattered into s points in the detector plane. Therefore the incident beam disk leads to s disks in the detector plane, part of which may lie outside the angular span captured on the camera. Simulating a CBED pattern requires one SBW simulation per incident beam vector in the disk. A finite angular resolution $\Delta\theta$ is used to sample the disk. The angular relationship between each point in a CBED disk is calibrated once for the whole pattern, yielding the incident beam tilts for every point in the pattern [5]. The incident beam tilt is both a member of P_{all} and a known value from a calibrated CBED pattern [5,6]. We use the CBED pattern to determine our updates for unknown parameters $Q_{\text{all}} \subseteq P_{\text{all}}$ by calculating parameter gradients for multiple incident beam tilts from the pattern and applying Rprop to the median calculated gradients, thus providing updates for Q_{all}.

In a realistic CBED pattern, many reflections will be excited, but only a few will be measured, forming a subset M of measured reflections. The unmeasured reflections must be included, including in the backpropagation, because they affect the observed scattering behavior. Because no intensity information is recorded for the unmeasured reflections, the mismatch error E_k can be calculated only for $k \in M$. Conversely, if $k \notin M$, E_k cannot be calculated and should not influence the ANN weight updates. Setting $E_k = 0$ if $k \notin M$ means that the unmeasured intensity at the detector does not directly contribute to the ANN weight updates, but the reflections are still preserved in the simulation. Changing the length of U to drop those reflections is precluded by the matrix-vector multiplication in Eq. (3).

III. RESULTS AND ANALYSIS

Now, we apply our specimen-parameter retrieval algorithm to simulated CBED patterns, retrieving depth-dependent, "third-dimension" parameters. We retrieve third-dimensional information for three sets of scenarios. The first set of scenarios (T_{10}, T_{20}, and T_{40}) assumes that all simulated CBED disks are measured in the experimental patterns. The second set of scenarios assume that fewer CBED disks are measured than simulated. The third set of scenarios take the second set and add noise. Our target in this work is the specimen that yielded the asymmetric CBED pattern in Fig. 2(a), which has only one type of third-dimensional variation: α_x, the specimen tilt in the [001] direction relative to the zone axis.

A. All disks

In this section, we examine optimization scenarios for three different sets of Q_{all}. For optimization, we use a square (20×20) point subarea of the reciprocal-space CBED disks, with each point 0.01° apart, so $p_p = 400$ different incident beam tilts are considered. The upper-left corner of the square is at the disk center. For these simulations, we include intensity data for all 197 simulated reflections, not just the subset shown in Fig. 2. In scenario T_{10}, the specimen is equally divided into ten 10 nm layers, and Q_{all} only consists of α_x for each layer. The mean value of $\alpha_x = -0.097^\circ$ is used as the starting value. Scenario T_{20} is as scenario T_{10}, but equally divided into twenty 5-nm layers. Scenario T_{40} is as scenario T_{20}, but Q_{all} consists of α_x and also α_y for each layer, the specimen tilt in the $[\bar{1}10]$ direction relative to the zone axis (starting value and true value 0.00°). These scenarios have respectively 10, 20, and 40 parameters in Q_{all}.

Figure 3 shows our algorithm successfully retrieving third-dimension parameters at 5- and 10-nm depth resolution. Since we include only the zero-order-Laue-zone, we retrieve third-dimension information without requiring higher-order-Laue-zone information (cf. Ref. [39]). Figure 3(a) also shows that adjacent layers (e.g., 6, 7) with the same orientation have different convergence behavior. Abrupt parameter shifts (e.g., iteration 20, layers 1 and 6) are due to the Rprop algorithm, as detailed above.

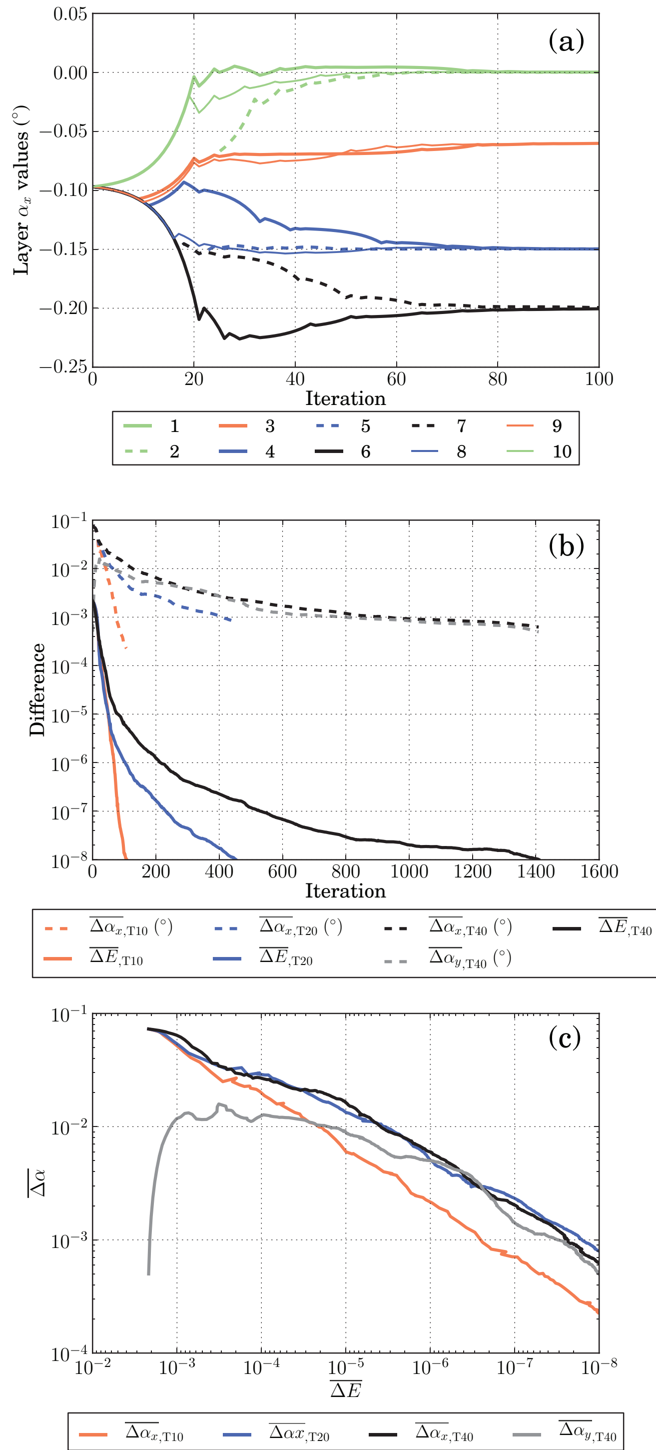

FIG. 3. (Color online) Results from scenarios T_{10}, T_{20}, and T_{40} (see text) for third-dimension parameter reconstruction in the all-disk case. (a) Scenario T_{10}: α_x layer-by-layer results for the first 100 iterations, showing retrieval of correct values. (b) All three scenarios: Mean intensity mismatch $\overline{\Delta E}$ taken over all p_p and mean parameter errors $\overline{\Delta\alpha_x}, \overline{\Delta\alpha_y}$ per iteration. (c) All three scenarios: Parameter error as a function of intensity mismatch, yielding detection thresholds.

Figure 3(b) shows mean intensity mismatch $\overline{\Delta E}$ (where $\Delta E = \sum_k E_k$ and the mean is taken over all $p_p = 400$ points) and mean parameter errors over all layers of the total set Q_{all} [$\overline{\Delta\alpha_x} = \mathtt{mean}(|\alpha_{x,Q} - \alpha_{x,\mathrm{true}}|)$ for scenarios T_{10} (light orange), T_{20} (dark blue), T_{40} (black) and $\overline{\Delta\alpha_y} = \mathtt{mean}(|\alpha_{y,Q} - \alpha_{y,\mathrm{true}}|)$ for scenario T_{40} (grey)] as a function of iteration between the first parameter update and convergence at $\overline{\Delta E} < 10^{-8}$. The $\overline{\Delta\alpha_y}_{,40}$ trajectory arises from the starting values being identical to the true values, and the calculation initially departing from those values but returning to converge on the true values. Increasing the number of parameters in Q_{all} leads to slower convergence. To reach $\overline{\Delta E} < 10^{-8}$, scenarios T_{10}, T_{20}, and T_{40} take 108, 454, and 1410 iterations, respectively, so doubling the number of parameters more than doubles the number of iterations until convergence. Time-to-convergence is dependent on the number of parameters and layer properties, but other third-dimension parameters and configurations may have different convergence characteristics.

Figure 3(c) shows the same data in Fig. 3(b) regraphed for mean parameter error $\overline{\Delta\alpha}$ as a function of mean intensity mismatch $\overline{\Delta E}$. The $\overline{\Delta\alpha_y}_{,40}(\overline{\Delta E})$ trajectory results from the same source as that in Fig. 3(b), noted above. When optimizing against experimental data, noise limits the best possible $\overline{\Delta E}$ value. Figure 3(c) thus demonstrates, in principle, how precisely $\overline{\Delta\alpha}$ can be known for noisy data, providing a quality-of-fit metric.

The quality of parameter fit as a function of $\overline{\Delta E}$ also appears to be dependent on layer properties. In Fig. 3(c), scenario T_{10} follows a different trajectory than scenarios T_{20} and T_{40}, and is approximately an order of magnitude more precise for a given $\overline{\Delta E}$. T_{20} and T_{40} follow similar trajectories and have the same layer properties, but different Q_{all}. Thus the quality of fit for these scenarios appears more dependent on layer properties (e.g., layer thickness, number of layers) than the number of parameters in Q_{all}. Because changing the total specimen thickness would lead to a different CBED pattern, disentangling the individual influences of layer thickness and the number of layers without changing the CBED pattern would require heterogeneous layer thicknesses, which are not considered here.

B. Realistic disk subsets without noise

Now, we show how our algorithm performs when only a subset of simulated disks are measured, corresponding to the realistic experimental situation where only a few disks might be measured. This section uses the same simulation conditions and specimen as scenario T_{10} except, in this subsection, each disk in the subset is sampled using a (13 × 13)-point square grid with a 0.05° grid spacing; as above, the upper-left corner is at the orientation shown as the center of the pattern in Fig. 2. This different grid is used to demonstrate our algorithm's versatility.

Six different disk subsets were chosen, as shown in Table I, ranging from 1 to 9 measured reflections; the full all-disk set was also considered. The all-197-reflections-measured case is also considered, for comparison. Figure 4(a) shows the convergence behavior as a function of simulation epoch in the same way as Fig. 3(b), and Fig. 4(b) shows the quality of fit in the same way as Fig. 3(c). Figure 4 shows that our algorithm functions even when only one disk is measured, but that measuring more disks generally yields a faster and better fit. When operating our algorithm on other CBED patterns or using other sampling grids, some of these specifics

will likely change. However, these results do demonstrate that our algorithm can successfully retrieve third-dimensional information if only a few CBED disks are measured. In the next section, we increase the realism of our simulations by including noise.

C. Realistic disk subsets with noise

Now, we consider how noise in the recorded intensities affects our third-dimension information retrieval algorithm. Here, Poisson noise is added to the "target" data, so that the total number of electrons q along a given incident beam vector over a given time follows a Poisson distribution. We can choose different values of q, which correspond to different electron doses; higher q means higher electron dose and less noise. For example, if the normalized noise-free intensity for a given pixel in a CBED disk is 0.3, and $q = 10^4$, then the noisy intensity at that pixel would be the result pulled from a Poisson distribution centered at 3×10^3, which is then divided by q to rescale the intensities. To do this, we use the numpy.random.poisson function provided in the NUMPY PYTHON module [45].

Example CBED patterns with different noise levels can be seen in Fig. 5, ranging from 10^2 to 10^5 electrons in the incident beam per detector pixel. For experimental data, the maximum usable dose would be influenced by radiation damage and contamination effects. Additionally, doubling the radius of the CBED disk on the camera would quadruple the number of incident-beam electrons required for the same noise level per pixel. Thus, if the specimen is beam-sensitive or contaminates under the beam, it would be useful to know how beam dose affects the quality of the information retrieved using our algorithm.

In these simulations, only the five lowest-order reflections are measured, corresponding to the "5" case in Table I and Fig. 4. To explore the effects of noise, two parameters are varied: number of electrons q and number of reciprocal-space points used p_p. Intuitively, either increasing p_p or increasing q should improve the quality of fit. p_p was varied by using a square grid, as above. The four corners of the grid were fixed (and included) at $k_1 = 0x + 0y$, $k_2 = 1x + 0y$, $k_3 = 0x + 1y$, and $k_4 = 1x + 1y$, where k_1 is the beam tilt in the center of the pattern in Fig. 2(a), and x and y are 0.6° in the [001] and $[\bar{1}10]$ directions, respectively. The area bounded by this square was sampled with equal space between the grid points, and included the points on all four edges of the square. For example, a $p_p = 625$-point simulation used a square grid of (25×25) points with 0.025° grid spacing, and a $p_p = 49$-point simulation used a square grid of (7×7) points with 0.1° grid spacing; in both these simulations, k_1, k_2, k_3, and k_4 are included. Therefore the same reciprocal-space region was included in all simulations, but sampled differently.

TABLE I. List of limited-reflection subsets used in Fig. 4. As is conventional, single reflections are in [brackets] and families of reflections are in ⟨angle brackets⟩ with the number of family members indicated after.

	Refl.		Refl.
1a:	$[1\bar{1}1]$	5:	{[000], ⟨111⟩ (4×)}.
1b:	$[00\bar{2}]$	9:	{[000], ⟨111⟩ (4×),
2a:	[000], $[1\bar{1}1]$		⟨002⟩ (2×), ⟨220⟩ (2×)}.
2b:	$[1\bar{1}1]$, $[\bar{1}1\bar{1}]$	197:	All 197 sim. refl.

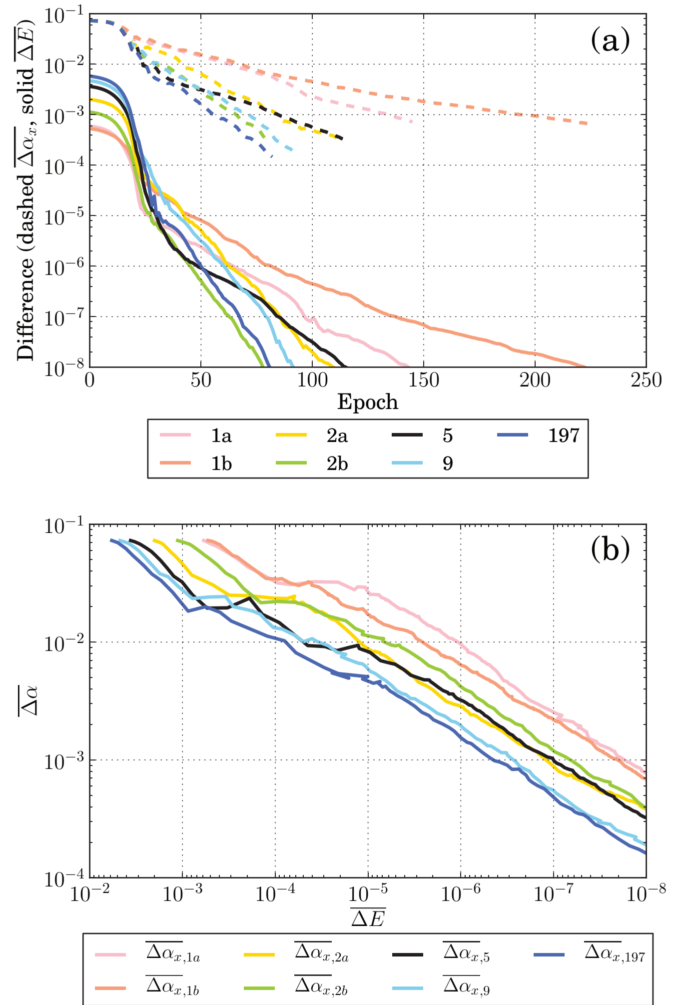

FIG. 4. (Color online) Results for disk subsets, graphed in the same manner as Figs. 3(b) and 3(c). These simulations all use a (13×13)-point grid with a 0.05° grid spacing. (a) shows mean parameter error $\overline{\Delta\alpha_x}$ (dashed lines) and mean intensity mismatch error $\overline{\Delta E}$ (solid lines) plotted as a function of simulation epoch (i.e., iteration). (b) shows $\overline{\Delta\alpha_x}$ as a function of $\overline{\Delta E}$, which is the parameter fit quality at a given intensity mismatch. The key for which subsets of reflections are used is in Table I; the number indicates the number of included reflections.

Figure 6 shows both the convergence behavior and the quality-of-fit results when noise is included, for a range of different noise levels ($q = 10^3$ to 10^5 with $p_p = 625$) and, for a set number of electrons, a range of different samplings ($p_p = 49$ to 921 with $q = 10^4$). The different noise levels, shown with the black and thin grey lines, substantially influence the quality of the reconstruction and the number of epochs required to reach convergence. Using an order of magnitude more electrons leads to about an order of magnitude better intensity mismatch, and about half an order of magnitude better parameter error. Thus, even with noisy input data, our algorithm is still able to retrieve third-dimension information.

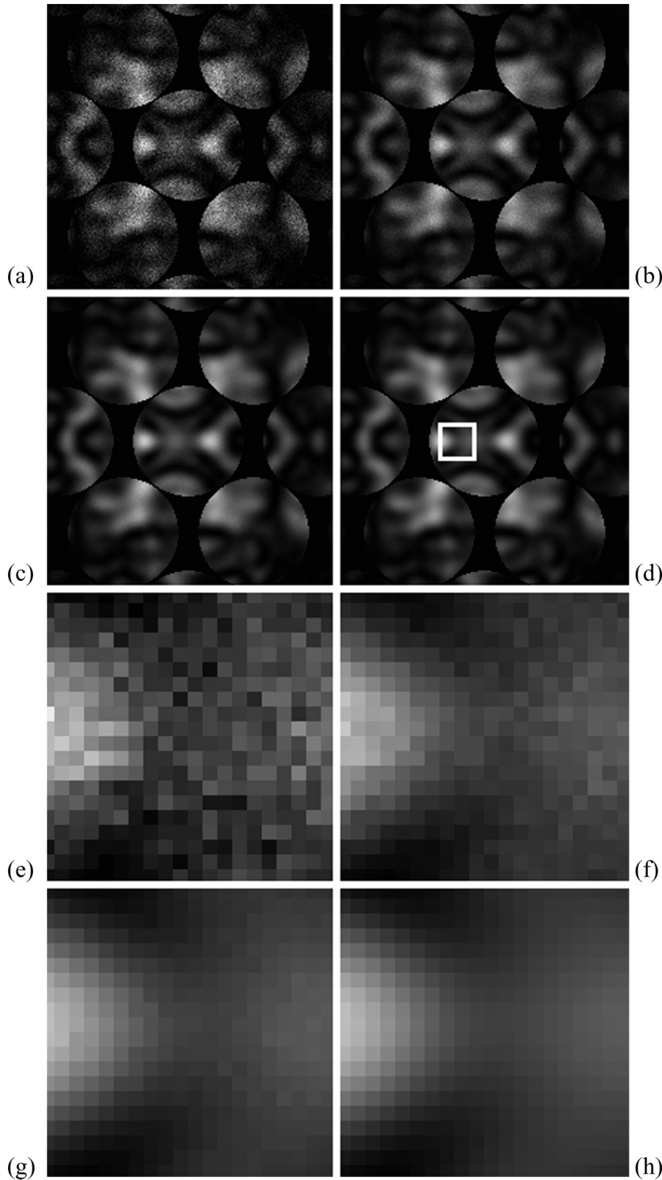

FIG. 5. CBED patterns with Poisson intensity noise (see text), corresponding to an incident beam intensity of (a) 10^2, (b) 10^3, (c) 10^4, and (d) 10^5 electrons per pixel. Specimen used is the same as that in Fig. 2(a). The inside edge of the white box added to (d) shows the region used in (e)–(h). (e)–(h) A 20×20-pixel (dot pitch 0.01°) excerpt from (a)–(d), showing the central disk, between the left maximum and the central cross. This is not the area used for optimization, but simply included to highlight the noise levels.

For a fixed noise level ($q = 10^4$ electrons), Fig. 6(a) shows that the $p_p = 49$ and $p_p = 169$-reciprocal-space-point scenarios yield less accurate fits than the $p_p = 625$ or $p_p = 921$ scenarios. For this specimen and noise level, therefore, $p_p = 49$ or $p_p = 169$ may be too few data points to optimally fit the noisy pattern. The $p_p = 625$ scenario retrieves slightly more accurate $\overline{\Delta\alpha}$ values than the $p_p = 921$ scenario, which indicates that more points does not automatically lead monotonically to an improved fit. Some of the differences between the scenarios may also be due to the random noise, so they may yield different results if run again with different values from the same Poisson distributions.

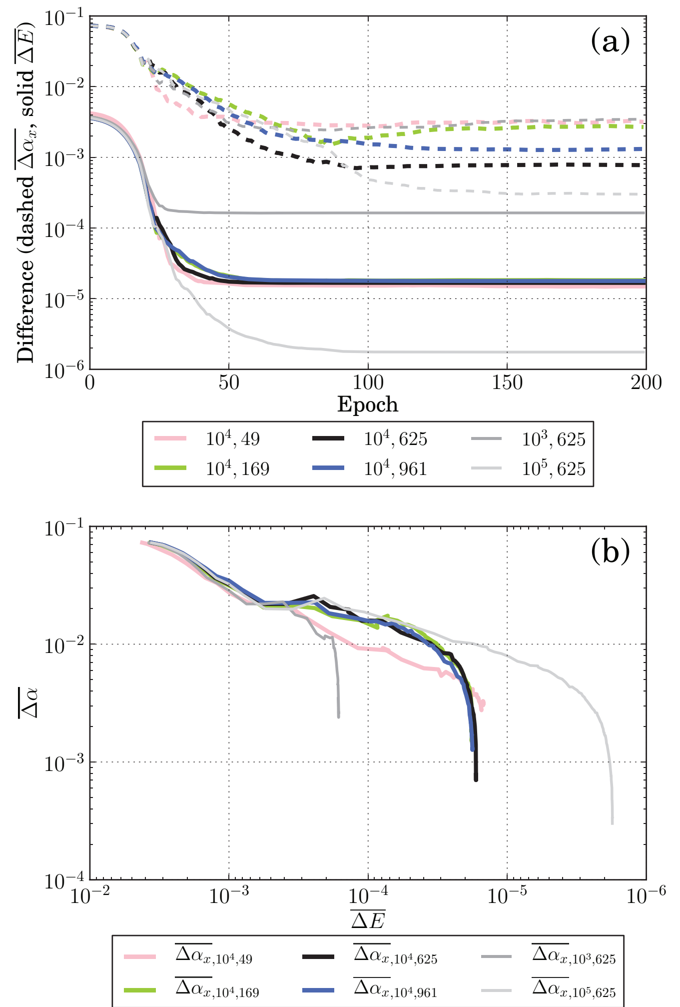

FIG. 6. (Color online) Results including noise and using only a few measured CBED disks, graphed in the same manner as Fig. 4. For an explanation for the "waterfall" curves in (b), see text.

The minimum $\overline{\Delta\alpha}$ values noted in Fig. 6(b) are the best $\overline{\Delta\alpha}$ values reached. If the minimum $\overline{\Delta\alpha}$ is reached and then the $\overline{\Delta\alpha}$ values get worse at the same $\overline{\Delta E}$, then the lines in Fig. 6(b) would be retraced. Because Fig. 6(a) shows some of these scenarios converging on less accurate values than the best-fit-reached, interpreting the $\overline{\Delta\alpha}$ minimums in Fig. 6(b) requires checking with Fig. 6(a) to ensure that the minimum is also the converged value. Thus realistic CBED patterns may have improved third-dimension parameter retrieval from increasing the number of reciprocal-space points used to fit, but adding more points is not necessarily guaranteed to improve the quality of fit. Additionally, the intensity noise may change convergence behavior and parameter values for different runs with the same noise distribution.

At first glance, Fig. 6(b) appears counterintuitive, but is likely due to how Rprop operates on a set of p_p different incident beam tilts. The mean parameter error can still change even when the mean intensity mismatch is constant (the "waterfall" feature on the graph). The $p_p = 49$ simulation

uses the fewest points and does not exhibit this behavior, but the others do. There is a possible explanation: while the *mean* intensity mismatch taken over many points is not improving, our optimization algorithm is using many individual points with Rprop to determine parameter updates. As detailed above, our algorithm, via Rprop [14], uses the sign of the median gradient across all points to determine parameter updates. While both signal and noise contribute to the mismatch, the signal will be correlated across multiple points, because it is due to the underlying specimen, while the noise is uncorrelated. In effect, the parameter updates are driven by the two-option vote of a majority of points on the sign of the gradient (either positive or negative). Because equal numbers of random positive and negative votes cancel each other out, this two-option, majority-voting mechanism might be expected to be somewhat robust against uncorrelated noise influencing a correlated signal. The $p_p = 49$ simulation appears to have too few points for this robustness to function; thus, the noise can override the signal if there are too few points included in the simulation. However, the mean intensity mismatch would still be noise-limited, and the mean parameter error might not necessarily improve if too few points are used.

In short, our algorithm demonstrates the ability to retrieve third-dimensional information even from only a few CBED disks with realistic noise. In the presence of noise, our algorithm appears to generally benefit from sampling more than a minimum number of points in the CBED pattern. Interestingly, the mean parameter error can still change even when the mean intensity mismatch stays constant, possibly because the latter is uncorrelated noise against which our parameter update mechanism could be resistant.

IV. CONCLUSIONS

In this work, we demonstrate the successful extraction of "third-dimension" (depth-dependent) materials properties quantitatively from TEM diffraction data including multiple scattering. Our algorithm reformulates the stacked-Bloch-wave TEM simulation algorithm as an artificial neural network (ANN) and applies backpropagation of error and resilient backpropagation to the ANN to retrieve third-dimension specimen parameters. This algorithm was used to retrieve the third-dimension specimen orientation with 5- and 10-nm resolution from simulated convergent-beam electron diffraction (CBED) data. In principle, this retrieval algorithm can be extended to any or all parameters affecting the electron beam in the sample, yielding depth-dependent information from a single TEM CBED pattern. This algorithm also works when using a realistic number of measured CBED disks and noisy data. Future work should examine heterogeneous layer thicknesses and other optimization characteristics to establish the general precision of third-dimension parameter determination, as well as further exploration of our algorithm's convergence properties on noisy data. This third-dimension optimization should also be extended to additional parameters, including scattering and structure factors, lattice parameter and strain, chemical composition, Debye-Waller factor, and specimen thickness, and should also be compared with experimental data, including CBED patterns, large-angle rocking-beam electron diffraction (LARBED) patterns [46] or dark-field image tilt series.

ACKNOWLEDGMENTS

The authors acknowledge funding from the German Research Foundation (Grant No. KO 2911/7-1) and the Carl Zeiss Foundation.

[1] M. Knoll and E. Ruska, Z. Phys. **78**, 318 (1932).
[2] D. B. Williams and C. B. Carter, *Transmission Electron Microscopy*, 2nd ed. (Springer, New York, 2009).
[3] P. E. Batson, N. Dellby, and O. L. Krivanek, Nature (London) **418**, 617 (2002).
[4] H. Bethe, Ann. Phys. (Berlin) **392**, 55 (1928).
[5] J. C. H. Spence and J. M. Zuo, *Electron Microdiffraction* (Plenum Press, New York, NY, 1992).
[6] C. T. Koch and J. C. H. Spence, J. Phys. A **36**, 803 (2003).
[7] P. A. Midgley, M. Weyland, J. M. Thomas, and B. F. G. Johnson, Chem. Commun. **2001**, 907 (2001).
[8] K. J. Batenburg and J. Sijbers, in *IEEE International Conference on Image Processing, 2007, ICIP 2007, San Antonio, TX, 2007* (IEEE, Piscataway, NJ, 2007), Vol. 4, pp. IV-133–IV-136.
[9] A. Alpers, R. J. Gardner, S. König, R. S. Pennington, C. B. Boothroyd, L. Houben, R. E. Dunin-Borkowski, and K. Joost Batenburg, Ultramicroscopy **128**, 42 (2013).
[10] P. D. Nellist, G. Behan, A. I. Kirkland, and C. J. D. Hetherington, Appl. Phys. Lett. **89**, 124105 (2006).
[11] E. C. Cosgriff, A. J. D'Alfonso, L. J. Allen, S. D. Findlay, A. I. Kirkland, and P. D. Nellist, Ultramicroscopy **108**, 1558 (2008).
[12] P. Wang, G. Behan, A. I. Kirkland, P. D. Nellist, E. C. Cosgriff, A. J. D'alfonso, A. J. Morgan, L. J. Allen, A. Hashimoto, M. Takeguchi, K. Mitsuishi, and M. Shimojo, Ultramicroscopy **111**, 877 (2011).
[13] R. Rojas, *Neural Networks: A Systematic Introduction* (Springer Verlag, Berlin, 1993).
[14] M. Riedmiller and H. Braun, in *IEEE International Conference on Neural Networks, 1993, San Francisco, CA, 1993* (IEEE, Piscataway, NJ, 1993), Vol. 1, pp. 586–591.
[15] K. Ishizuka and N. Uyeda, Acta Cryst. A **33**, 740 (1977).
[16] K. Ishizuka, Acta Cryst. A **38**, 773 (1982).
[17] C. H. Mac Gillavry, Physica **7**, 329 (1940).
[18] K. Kambe, Ultramicroscopy **10**, 223 (1982).
[19] P. B. Hirsch, A. Howie, R. Nicholson, D. W. Pashley, and M. J. Whelan, *Electron Microscopy of Thin Crystals*, 2nd ed. (Robert E. Krieger Publishing Co., Malabar, FL, 1977).
[20] M. Saunders, P. Midgley, R. Vincent, and J. Steeds, J. Electron Microscopy **45**, 11 (1996).
[21] J. M. LeBeau, S. D. Findlay, L. J. Allen, and S. Stemmer, Phys. Rev. Lett. **100**, 206101 (2008).
[22] A. Thust, Phys. Rev. Lett. **102**, 220801 (2009).

[23] J. M. Zuo and J. Spence, Ultramicroscopy **35**, 185 (1991).
[24] J. M. Zuo, P. Blaha, and K. Schwarz, J. Phys.: Cond. Mat. **9**, 7541 (1997).
[25] M. Hytch and W. Stobbs, Ultramicroscopy **53**, 191 (1994).
[26] A. Howie, Ultramicroscopy **98**, 73 (2004).
[27] F. F. Krause, K. Müller, D. Zillmann, J. Jansen, M. Schowalter, and A. Rosenauer, Ultramicroscopy **134**, 94 (2013).
[28] R. Vincent, T. D. Walsh, and M. Pozzi, Ultramicroscopy **76**, 125 (1999).
[29] D. Jacob and A. Lefebvre, Ultramicroscopy **96**, 1 (2003).
[30] R. S. Pennington, F. Wang, and C. T. Koch, Ultramicroscopy **141**, 32 (2014).
[31] D. J. Eaglesham, C. J. Kiely, D. Cherns, and M. Missous, Phil. Mag. A **60**, 161 (1989).
[32] T. Yamazaki, K. Watanabe, K. Kuramochi, and I. Hashimoto, Acta Cryst. A **62**, 233 (2006).
[33] M. Ohtsuka, T. Yamazaki, I. Hashimoto, and K. Watanabe, Acta Cryst. A **65**, 135 (2009).
[34] W. Van den Broek and C. T. Koch, Phys. Rev. Lett. **109**, 245502 (2012).
[35] W. Van den Broek and C. T. Koch, Phys. Rev. B **87**, 184108 (2013).
[36] A. Béché, J. L. Rouvière, J. P. Barnes, and D. Cooper, Ultramicroscopy **131**, 10 (2013).
[37] A. Lubk, E. Javon, N. Cherkashin, S. Reboh, C. Gatel, and M. Hÿtch, Ultramicroscopy **136**, 42 (2014).
[38] F. Houdellier, C. Roucau, L. Clément, J. L. Rouvière, and M. J. Casanove, Ultramicroscopy **106**, 951 (2006).
[39] F. Houdellier, A. Altibelli, C. Roucau, and M. J. Casanove, Ultramicroscopy **108**, 426 (2008).
[40] L. M. Peng, G. Ren, S. L. Dudarev, and M. J. Whelan, Acta Cryst. A **52**, 257 (1996).
[41] M. Schowalter, A. Rosenauer, J. T. Titantah, and D. Lamoen, Acta Cryst. A **65**, 5 (2008).
[42] D. Lide, *CRC Handbook of Chemistry and Physics*, 89th Edition (CRC Press, Boca Raton, 2008).
[43] D. M. Bird and Q. A. King, Acta Crystallographica A **46**, 202 (1990).
[44] P. Rez, Phys. Status Solidi A **55**, K79 (1979).
[45] T. E. Oliphant, Comput. Sci. Eng. **9**, 10 (2007).
[46] C. T. Koch, Ultramicroscopy **111**, 828 (2011).