Nuclear Instruments and Methods in Physics Research B 574 (2026) 166076

Contents lists available at ScienceDirect

Nuclear Inst. and Methods in Physics Research, B

journal homepage: www.elsevier.com/locate/nimb

Full length article

AutoNRA: A high-throughput parallel tool for self-consistent ion beam analysis☆

K.F. Muzakka [a,*], S. Möller [a,b], S.A. Afzal [a], M. Bram [a], M. Finsterbusch [a]

[a] Forschungszentrum Jülich GmbH, Institute of Energy Materials and Devices, Materials Synthesis and Processing (IMD-2), Jülich, Germany
[b] Aachen Ion Beams UG, Aachen, Germany

ARTICLE INFO

Keywords:
Ion beam analysis
Rutherford backscattering spectrometry
Nuclear reaction analysis

ABSTRACT

We present AutoNRA, a high-throughput framework for self-consistent ion beam analysis (IBA). Developed to meet the growing demand for automated processing of large-scale datasets, AutoNRA enables parallel analysis of spectra from thousands of sample points, each potentially containing multiple IBA measurements such as Rutherford Backscattering Spectrometry (RBS) and Nuclear Reaction Analysis (NRA). By exploiting multiprocessing, it fully utilizes available CPU cores and achieves speed-ups of up to two orders of magnitude compared to conventional sequential SIMNRA workflows. This acceleration makes it particularly suited for spatially resolved applications, including three-dimensional compositional tomography.

AutoNRA ensures model consistency by maintaining shared parameters across spectra from the same sample point while performing independent, parallel evaluations across different points. The framework also provides a model-agnostic machine learning interface, allowing seamless integration of surrogate models for accelerated spectrum prediction and adaptive optimization. Its performance and scalability are demonstrated on BZCY721($BaZr_{0.7}Ce_{0.2}Y_{0.1}O_{3-\delta}$) sample fabricated by cold sintering, where AutoNRA enabled quantitative mapping of elemental distributions and detection of residual heavy water within the structure.

1. Introduction

Modern ion beam analysis (IBA) techniques such as Rutherford Backscattering Spectrometry (RBS), Nuclear Reaction Analysis (NRA), Particle-Induced Gamma-ray Emission (PIGE), and Particle-Induced X-ray Emission (PIXE) constitute a family of powerful and complementary methods for quantitative elemental and structural characterization of materials. Owing to their sensitivity to elemental composition and depth distribution, IBA methods have been employed for decades across a wide range of disciplines, including semiconductor device engineering, energy materials research, thin-film technology, nuclear materials, geosciences, and cultural heritage studies. Their unique ability to provide non-destructive, high-depth-resolution profiles of both light and heavy elements makes IBA indispensable in both fundamental research and applied materials development [1].

With continuing advances in accelerator technology and detector systems, IBA has evolved from isolated point measurements to high-throughput, spatially resolved, and even tomographic applications [2]. In modern workflows, three-dimensional elemental imaging is no longer obtained from a handful of characteristic spectra; instead, it requires the acquisition and analysis of thousands of spectra across extended sample regions. Each spatial pixel is associated with a full spectrum that must be quantitatively interpreted by fitting an IBA model, a procedure that is both computationally demanding and time consuming. However, it must be noted that the practical feasibility of such pixel-wise analysis depends on the availability of sufficient counting statistics at each pixel within realistic acquisition times. This requirement can be met either by selecting reaction channels with relatively high cross sections or by employing instrumental configurations, such as high-efficiency detectors, large solid angles, and increased beam currents, that provide adequate count rates.

For non-depth-sensitive IBA techniques such as PIXE, such large-scale, pixel-wise analysis is already established, since the extraction of elemental composition from PIXE spectra can in many cases be performed rapidly and with limited model complexity. By contrast, depth-sensitive IBA techniques such as RBS and NRA pose a qualitatively harder challenge: here, each individual spectrum encodes not only which elements are present, but also how they are distributed as a function of depth. As a consequence, every spectrum requires the solution of an inverse problem that links the measured yield to an unknown depth profile, under physical constraints such as stopping power, cross

☆ This article is part of a Special issue entitled: '2025 IBA – PIXE' published in Nuclear Inst. and Methods in Physics Research, B.

* Corresponding author.
E-mail address: k.muzakka@fz-juelich.de (K.F. Muzakka).

https://doi.org/10.1016/j.nimb.2026.166076
Received 25 November 2025; Received in revised form 4 February 2026; Accepted 2 March 2026
Available online 6 March 2026

sections, and detector geometry. Large-area, high-resolution mapping with depth-sensitive methods therefore demands the execution of a very large number of high-dimensional fits, resulting in a substantial computational burden.

To address the associated computational cost, clustering-based and machine-learning-driven approaches have been explored to identify representative spectra or to approximate the inversion process [3–5]. While these methods can significantly reduce analysis time, they generally rely on data reduction, surrogate models, or user-defined training and classification criteria, which may lead to a loss of spatial detail and can introduce systematic biases that are difficult to quantify within a strict physical modeling framework. In contrast, recent advances in experimental hardware, including high-throughput detector systems and increased usable beam currents, together with substantial gains in computing performance (multi-core CPUs and GPU acceleration) now render fully pixel-wise, physically constrained analysis increasingly practical, without requiring compromises in spatial resolution or model fidelity.

Traditionally, IBA spectral analysis has relied on sequential fitting workflows using simulation programs such as SIMNRA [6]. In such approaches, experimental spectra are iteratively compared with simulated spectra, and model parameters are adjusted until the best-fit elemental depth profiles (EDPs) are obtained. While robust and extensively validated, sequential SIMNRA analysis becomes increasingly impractical as dataset sizes grow. For a relatively complex case, a single SIMNRA simulation typically requires on the order of one second, and the iterative fitting of a single spectrum can involve 10^4 or more simulation calls. Consequently, the analysis of a single measurement point may consume several hours of computation time. When extended to spatially resolved or tomographic datasets comprising thousands of measurement points, the computational burden rapidly escalates to weeks or even months. This bottleneck is further amplified in self-consistent analyses, where multiple spectra from the same point must be fitted simultaneously under shared physical constraints.

Efforts to alleviate these limitations have led to the development of more advanced analysis frameworks. MultiSIMNRA [7], for example, extends SIMNRA towards self-consistent, automated fitting of multiple spectra by linking shared parameters and coordinating several simulations within a common optimization loop. This significantly reduces manual interaction for multi-spectral or multi-geometry problems. However, such tools are typically geared towards batches of limited size and do not provide built-in support for genuinely large-scale studies involving thousands of measurement points, nor do they offer integrated tomographic visualization or interactive 3D exploration of fitted EDPs. Large mapping or tomography datasets still require substantial external scripting and post-processing to be handled efficiently.

More broadly, most existing IBA analysis tools [6–8] provide only partial support for automated, high-throughput workflows. With the notable exception of NDF DataFurnace [9], which implements a powerful global optimization framework, widely used codes such as SIMNRA were originally designed for manual, spectrum-by-spectrum analysis. As a result, IBA practitioners face a persistent trade-off between accuracy, throughput, and reproducibility. Manual fitting is not only time-consuming but also susceptible to user bias and inconsistent treatment of shared parameters across related spectra. These limitations significantly constrain the scope of modern IBA studies and prevent full exploitation of high-throughput experimental capabilities and tomographic measurement strategies.

To address these challenges, we present AutoNRA, a high-throughput batch analysis framework for self-consistent IBA. AutoNRA is designed with modularity and scalability at its core to utilize all available computational resources. The framework enforces parameter consistency — that is, self-consistency — across spectra associated with the same measurement point, while still permitting independent evaluation of different points. Owing to its architecture, AutoNRA is not only well suited for point-by-point evaluations but also enables applications such as spatially resolved depth profiling and tomographic reconstruction.

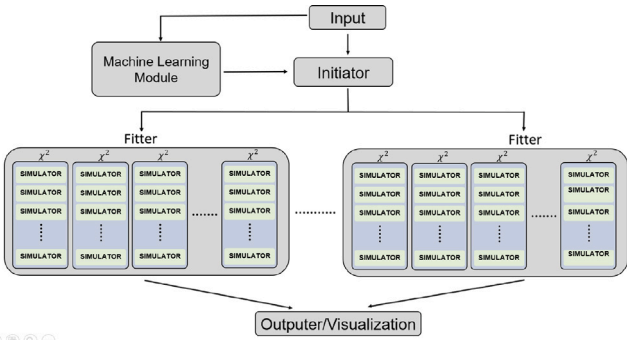


Fig. 1. Schematic representation of the AutoNRA framework, illustrating the interaction between the input, initiator, and multiple parallel fitters running SIMNRA simulations, followed by output generation and visualization.

Looking ahead, the emergence of scalable analysis frameworks opens new opportunities for integrating machine learning into ion beam analysis. With access to large, well-curated datasets of both simulated and experimental spectra, data-driven models can be trained to approximate or accelerate the inversion process from spectra to elemental depth profiles. These approaches have the potential to enable near real-time elemental imaging, mitigating the limitations of traditional iterative fitting methods and reshaping the role of IBA in materials science and related disciplines.

In this context, AutoNRA has been developed with machine learning (ML) applications in mind, allowing users to generate training data, train models, and perform inference within a single environment. The current version of AutoNRA already supports ML-based surrogate models that can entirely replace SIMNRA, offering significantly faster and highly parallelized IBA simulations, albeit with a slight trade-off in accuracy. Although still in its early stages, initial results from this surrogate-model approach are very promising and demonstrate the transformative potential of this approach.

2. Framework

AutoNRA is designed with modularity in mind, specifically to accommodate the use of different simulator engines that calculate the IBA spectrum given a set of elemental depth profiles and experimental setup parameters. Currently, two options are available: SIMNRA and an ML-based surrogate model. In this work, we focus on AutoNRA in the context of SIMNRA as the simulator. The application of AutoNRA with the ML-based surrogate model will be explored in a separate study.

The overall framework of AutoNRA is illustrated in Fig. 1. At its core, AutoNRA converts a single configuration file into a fully parallel and self-consistent analysis workflow capable of handling multiple spectra and various IBA methods. The process begins with a YAML (“YAML Ain’t Markup Language”)–based input file, a human-readable data-serialization format widely used for configuration due to its clarity and flexibility. This file defines the sample and layer structures, computational parameters, fitting configurations, and general application settings. During initialization, AutoNRA employs heuristic strategies or, optionally, an inverse machine-learning module to generate initial estimates of the elemental depth profiles. Although the ML component is currently implemented as a placeholder, it has been designed to support future integration of data-driven inversion techniques.

The fitting stage is managed by a parallelized engine, where each sample is assigned an independent fitter. Candidate solutions are evaluated in batches, and each batch is distributed across multiple simulator instances through a multithreaded backend that performs the IBA spectrum calculations. These multithreaded computations are implemented

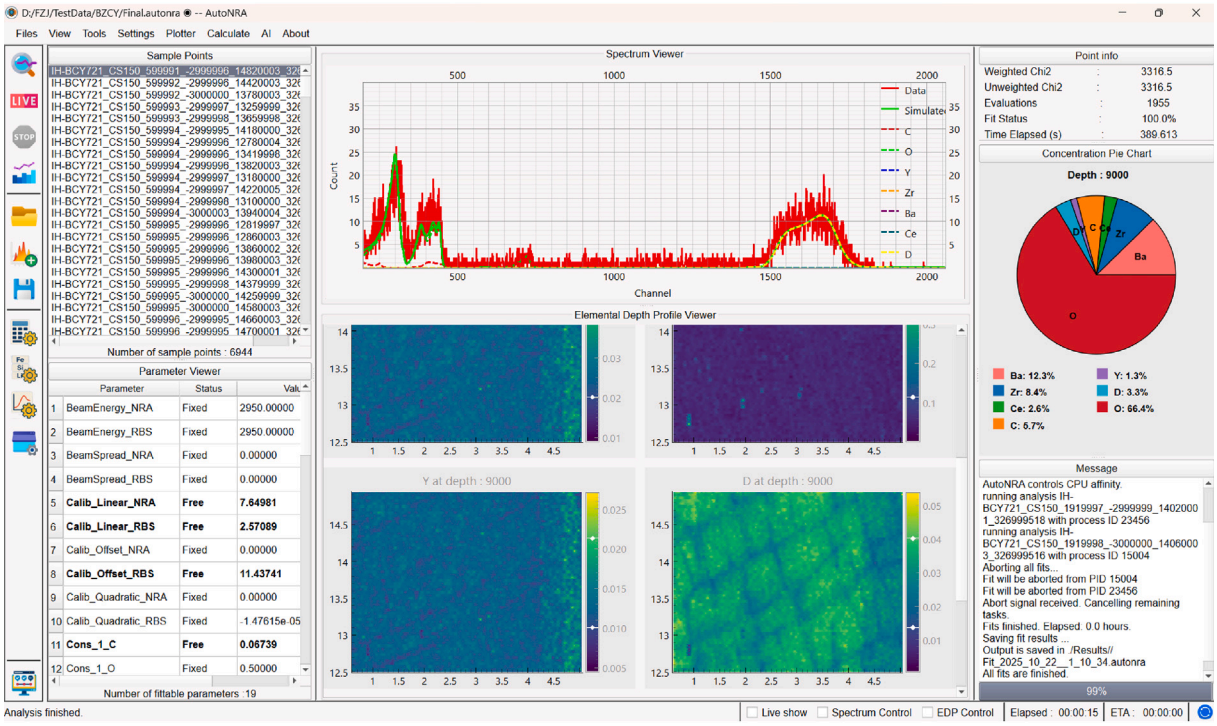


Fig. 2. AutoNRA's main graphical user interface (GUI).

as a dedicated `Cython` library, which interfaces with the graphical user interface (GUI) module for visualization and export of analysis results.

AutoNRA stores and exports all analysis data — including the application state, user settings, sample configurations, and fitting parameters — within a single YAML file bearing the `.autonra` extension. In addition to analysis results and general application configurations, the file structure comprises three primary blocks: *layer settings*, *calculation settings*, and *fit settings*.

The *layer settings* section defines the assumed sample configuration, including the number of layers, constituent elements or isotopes, and optional parameters such as porosity and roughness. Users may also specify a smooth parameterization for depth-dependent concentrations; in such cases, the layer is internally discretized into sublayers for simulation. The following parameterizations are currently supported:

- Constant:

$$n_i(y) = c_i \tag{1}$$

- Polynomial:

$$n_i(y) = \sum_{j=0}^{D} a_j y^j \tag{2}$$

- Erf–polynomial:

$$n_i(y) = A\left(1 + a\,\mathrm{erf}\left(\frac{y-b}{c}\right)\right)\left(1 + \sum_{j=1}^{D} a_j y^j\right) \tag{3}$$

Here, $n_i(y)$ is the unnormalized concentration of element i at depth $x = yd$, where d is the layer thickness. Concentrations are then normalized, $n_i(y) \to n_i(y)/\sum_k n_k(y)$, so that $\sum_k n_k(y) = 1$.

The *calculation settings* define the simulator configuration parameters, method-specific options for each IBA technique, and optimizer controls. For every IBA method, AutoNRA requires a reference file that specifies the experimental setup and simulator configurations. In the case of the SIMNRA simulator, this reference file provides a comprehensive list of candidate elements together with their associated cross sections. This list may be as extensive as necessary, since AutoNRA automatically excludes elements that are not present in the sample. Consequently, the same reference file can be reused across multiple analyses, provided the experimental configuration remains unchanged. During execution, AutoNRA overrides the layer and elemental definitions from the reference file and applies the customized layer model defined in the user configuration.

For simulations based on ML surrogate models, the reference file specifies the model required by AutoNRA to perform the ML-driven spectrum calculation. AutoNRA uses `ONNX Runtime` [10] to execute models exported in the `ONNX` format, allowing those originally developed in frameworks such as PyTorch or TensorFlow to be used in a backend-agnostic manner. Consequently, a model developed by one user can be reused by others as a black-box component, without requiring knowledge of its internal architecture.

In the case of the SIMNRA backend, AutoNRA communicates through the COM/OLE protocol. Low-level interprocess communication and lifecycle management are handled by a dedicated C++ control layer that safely schedules and synchronizes multiple simulator instances in parallel. For the ML-based surrogate backend, a serialized `ONNX` model is loaded into memory, and AutoNRA invokes `ONNX Runtime` to perform inference. Because this backend is inherently parallelizable — through batching and intra-/inter-op threading — no additional parallelization layer is necessary. The simulator module, which unifies both the SIMNRA and `ONNX Runtime` backends, exposes a consistent interface to higher-level components such as the fitting engine and workflow orchestrator, ensuring seamless integration regardless of the selected simulator.

The graphical user interface (GUI) is implemented using `PySide6` as the windowing framework, `PyQtGraph` for high-performance 2D spectral visualization, and OpenGL for hardware-accelerated 3D rendering. Users can interactively inspect experimental and simulated spectra, and visualize elemental depth profiles as well as tomographic reconstructions in real time (Fig. 2). Upon completion of an analysis, AutoNRA produces a single `.autonra` bundle containing the full configuration, fitted elemental depth profiles (EDPs), and per-spectrum statistics. For compatibility, individual SIMNRA `.xnra` files can also be exported for each point and method. With native Windows Explorer integration, `.autonra` files can be opened directly via double-click.

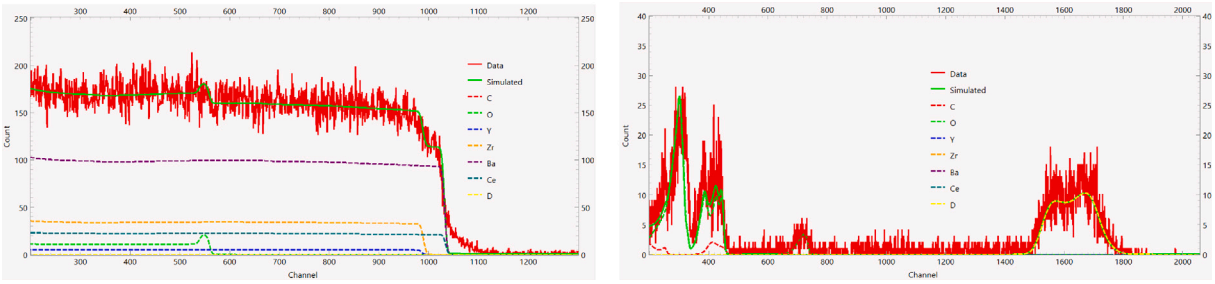


Fig. 3. Representative RBS (left) and NRA (right) spectra showing experimental data, SIMNRA simulations, and elemental contributions.

2.1. Optimization strategy

The optimization algorithm forms the core of AutoNRA, carrying out the numerical optimization of model parameters. For each spectrum s and each region of interest (ROI) R, the objective function is expressed as a weighted sum of χ^2 contributions. Multiple definitions of $\chi^2_{s,R}$ are available in AutoNRA:

- Neyman χ^2 :

$$\chi^2_{s,R,Neyman} = \sum_{i=1}^{N_R} \frac{(M_i - \mu_i)^2}{M_i} \tag{4}$$

- Pearson χ^2 :

$$\chi^2_{s,R,Pearson} = \sum_{i=1}^{N_R} \frac{(M_i - \mu_i)^2}{\mu_i} \tag{5}$$

- Poisson log likelihood ratio:

$$\chi^2_{s,R,PLLR} = 2\sum_{i=1}^{N_R} \left(\mu_i - M_i + M_i \log\left(\frac{M_i}{\mu_i}\right)\right) \tag{6}$$

- Gaussian log likelihood ratio:

$$\chi^2_{s,R,GLLR} = \sum_{i=1}^{N_R} \left(\frac{(M_i - \mu_i)^2}{\mu_i} + \ln\left(\frac{\mu_i}{\mu'_i}\right) - \frac{(M_i - \mu'_i)^2}{\mu'_i} \right), \tag{7}$$

$$\mu'_i \equiv \sqrt{M_i + 1/4} \tag{8}$$

- Mean squared error (MSE) :

$$\chi^2_{s,R,MSE} = \sum_{i=1}^{N_R} (\mu_i - M_i)^2 \tag{9}$$

Here, M_i and μ_i denote experimental and simulated counts at channel $i \in R$ while N_R is the number of data points in the ROI R. The Poisson log-likelihood ratio $\chi^2_{s,R,PRRL}$ is the most statistically sound as this corresponds to the normalized likelihood assuming that fluctuation in the sample are due to Poissonian statistics. When the measured counts are high enough, the Poisson distribution can be approximated by a Gaussian. In this case, the Gaussian log-likelihood ratio $\chi^2_{s,R,GLLR}$ can be used. In the large sample scenario, $\chi^2_{s,R,GLLR}$ reduce to Pearson $\chi^2_{s,R,Pearson}$[11]. However, from fitting point of view, while the $\chi^2_{s,R,Neyman}$ can leads to a biased fit [12], it is simpler in the functional form (only quadratic in μ), thus easier to fit. Thus in practice, $\chi^2_{s,R,Neyman}$ is almost always used and hence the default. Note that to ensure numerical stability, any zero-valued denominators in Eqs. (4) and (5), as well as terms involving logarithms or denominators in Eqs. (6) and (7), are replaced by unity during evaluation.

Given the weight w_R of the region of interest (ROI) R and the spectrum weight w_s, the total objective function is defined as

$$\chi^2 = \sum_s w_s \sum_R w_R \chi^2_{s,R}. \tag{10}$$

The weights w_s and w_R are user-settable parameters. The choice of weighting scheme can influence both the asymptotic convergence behavior of the fitting procedure and the bias of the estimated parameters [13]; therefore, careful consideration of these weights is required to ensure stable and physically meaningful solutions. Fitting is performed by minimizing the total χ^2 with respect to the elemental depth profile parameters, which are the parameters of the chosen layer parameterization, as well as additional nuisance parameters associated with the experimental setup, such as energy calibration, detector resolution (FWHM), and related instrumental factors.

In AutoNRA, for efficiency reasons, the fitting process is performed by calling the χ^2 function in batches. Instead of submitting individual simulation requests sequentially, AutoNRA groups multiple trial parameter sets into a single batch. These batches are then dispatched concurrently to multiple SIMNRA instances through a multithreaded execution model. By utilizing all available CPU cores, hundreds to thousands of simulations can be processed in parallel. Equally important, the optimization algorithm and its underlying library must natively support such batch calls. For this reason, we employ the Minion library [14], which inherently supports batch evaluations, enabling seamless integration between the optimization algorithm and the parallel SIMNRA backend.

The Minion library is a collection of state-of-the-art algorithms for bound-constrained or unconstrained single-objective optimization problems. Its backend is implemented in C++, providing high computational performance. It includes advanced Differential Evolution (DE) variants, particle swarm optimization (PSO), covariance matrix adaptation evolution strategy (CMAES), and standard algorithms such as Nelder–Mead, L-BFGS-B, and generalized simulated annealing. AutoNRA users have access to all these algorithms and can even chain up to four algorithms sequentially within a single optimization pipeline. For example, one may start with a Differential Evolution algorithm for global exploration, followed by L-BFGS-B or Nelder–Mead for local refinement.

3. Application

3.1. Data sets

As a representative test case, AutoNRA was applied to the analysis of Rutherford Backscattering Spectrometry (RBS) and Nuclear Reaction Analysis (NRA) spectra obtained from a BZCY721 ($BaZr_{0.7}Ce_{0.2}Y_{0.1}O_{3-\delta}$) specimen, a proton-conducting perovskite material widely used in protonic ceramic fuel cells and electrolysis cells [15,16]. The sample was fabricated via *cold sintering*, an energy-efficient consolidation process that densifies powders under high pressure in the presence of a transient liquid sintering aid, enabling diffusion at relatively low temperatures (typically below 300 °C) [17]. In the simplest case, pure water can be used as the sintering aid, mimicking hydrothermal processes observed in nature.

A key question, however, is whether residual sintering aids or other impurity compounds remain in the material in its as-cold-sintered state, and how these species are distributed throughout the microstructure.

To address this in the case of cold-sintered BZCY721 with water as the sintering aid, deuterium (D) was employed as a tracer for heavy water (D_2O), enabling quantitative mapping of its spatial distribution and concentration using IBA measurements.

The BZCY721 sample was prepared as follows. The starting powder was prepared in-house by mixing 50 g of precursor materials — $BaCO_3$ (Sigma-Aldrich, ≥99%), ZrO_2 (Sigma-Aldrich), CeO_2 (Sigma-Aldrich, ≥99%), and Y_2O_3 (Sigma-Aldrich, ≥99.99%) — according to the stoichiometric composition $Ba_{0.7}Zr_{0.2}Ce_{0.1}Y_{0.1}O_3$. The powder mixture was calcined at 1300 °C, resulting in three distinct phases: a major Zr-rich BZY91 ($BaZr_{0.9}Y_{0.1}O_{3-\delta}$) phase and minor phases of BZC64 ($BaZr_{0.6}Ce_{0.4}O_3$) and BZC28 ($BaZr_{0.2}Ce_{0.8}O_3$). Cold sintering was carried out under vacuum using a FAST/SPS apparatus (Field Assisted Sintering Technology/Spark Plasma Sintering; model HP-D5, FCT Systeme, Germany). The tooling and die were made of Mo-based TZM alloy (Plansee SE, Austria). A 2.5 g powder charge was hand-packed into a 13 mm diameter die. Subsequently, 5 wt.% D_2O was added using a micropipette to wet the powder surface. The mixture was pre-pressed for 60 s under a force of 6 kN to form a green compact. The sintering process was conducted at 150 °C under 400 MPa pressure, with a heating rate of 20 °C/min and a dwell time of 5 min.

The spectral data were acquired at the IBA facility of Forschungszentrum Jülich (FZJ) using a 2950 keV ^{3}He beam. Four detectors — PIGE, PIXE, NRA, and RBS — are installed in the μNRA end-station [18], allowing simultaneous collection of all modalities at each measurement point. The sample was scanned point-by-point by directing the ion beam onto a given location and then moving to the next. In total, 6944 points were measured across a $3 \times 5\,\mathrm{mm}^2$ area. In the present work, as a demonstration, only the RBS and NRA spectra are analyzed self-consistently.

The sample is modeled as a two-layer system consisting of a thin C,O-rich surface layer and a thicker subsurface layer containing C, O, Y, Zr, Ba, Ce, and D. The subsurface region is discretized into ten sublayers, and the elemental depth profiles (EDPs) are parameterized using an error-function polynomial model (see Eq. (3)). In this representation, each element is described by four parameters, A, a, b, and c. Because of the strong overlap between the cross sections of Ba, Ce, Zr, and Y, their EDP parameters are fixed according to the expected stoichiometric ratios. In contrast, all parameters associated with O, C, and D, as well as the calibration and ParticlesSr parameters (i.e., the product of the number of incident particles and detector solid angle), are left unconstrained.

For the definition of the region of interest (ROI), NRA spectra use channels 200–1800 (ROI length: 1600), while RBS spectra use channels 200–1200 (ROI length: 1000). The ROI weight w_R and the spectrum weight w_s are both set to unity. This choice is justified by the fact that the total counts in the RBS and NRA spectra are of the same order of magnitude; although the ROI lengths differ, this difference is not physically significant and while still facilitate direct comparison between the two techniques.

In total, 19 free parameters are optimized using the ARRDE [19] algorithm, with up to 1900 SIMNRA evaluations per fit.

The dataset comprises 13,888 spectra. Under a total evaluation budget of approximately 2.64×10^7 SIMNRA simulations, AutoNRA achieves an average processing throughput of 125 spectra/s, resulting in a total wall-clock fitting time of about 59 h on a Windows workstation with 64 CPU cores and a total of software 120 threads. Using a single-thread execution throughput of 2.8 spectra/s as a serial baseline, this corresponds to a parallel speedup of 44.6×. The resulting parallel efficiency is approximately 37% when considering 120 software threads, or about 70% when normalized by the number of physical CPU cores, indicating effective utilization of available hardware despite the overhead associated with COM/OLE-based external simulator control.

3.2. Optimization algorithm comparisons

As mentioned earlier, AutoNRA relies on the Minion library, which provides state-of-the-art implementations of many modern optimization algorithms. In this section, we compare the performance of several optimizers commonly used in the literature and in our previous work.

To evaluate optimizer performance under a fixed, low-cost evaluation budget, we benchmark eight algorithms on the BZCY721 dataset using $N = 31$ spectra randomly selected from the full dataset. The tested optimizers include three differential evolution (DE) variants—L-SHADE [20], JADE [21], and ARRDE [19]—one CMA-ES variant (BIPOP-CMAES [22]), one advanced PSO variant (DMS-PSO [23]), Generalized Simulated Annealing (GSA) [24]—also known as Dual Annealing (DA)—and two classical local optimizers: L-BFGS-B [25] and NELDER-MEAD (NM) [26].

The population size was initialized at 50 individuals for all DE and PSO variants. Each fit was initialized from parameters sampled randomly within broad bounds, except for calibration parameters, which were initialized close to their nominal values. Note that for standard DE algorithms such as L-SHADE and JADE, the recommended initial population size is typically large (often $> 10\times$ the dimension). However, in our case the maximum number of function evaluations is constrained to $N_{\max} = 1900$ for $D = 19$ free parameters. Using such large initial populations would leave too few generations for meaningful evolution, making these methods uncompetitive under our evaluation budget. ARRDE, by contrast, is explicitly designed to remain effective under low $N_{\max}$ conditions and therefore does not suffer from this limitation.

We report summary statistics of the final objective function values in Table 1, where lower values indicate better fits. Among the tested algorithms, NELDER-MEAD achieves the lowest overall χ^2 values, both in terms of mean and median performance. Its performance is closely matched by ARRDE, which attains nearly identical objective values and remains one of the most reliable population-based optimizers. Considering its strong performance under a low evaluation budget and its natural parallelizability as a population-based method, we therefore recommend ARRDE as the default global optimizer in AutoNRA.

3.3. Results

After the analysis, the mean value of the overall objective function across the scanned area is found to be 3281 ± 165. When decomposed by IBA technique, the average χ^2 contribution from RBS spectra (ROI length: 1000 channels, degrees of freedom: 981) is 1846 ± 125, while the corresponding contribution from NRA spectra (ROI length: 1600 channels, degrees of freedom: 1581) is 1434 ± 114.

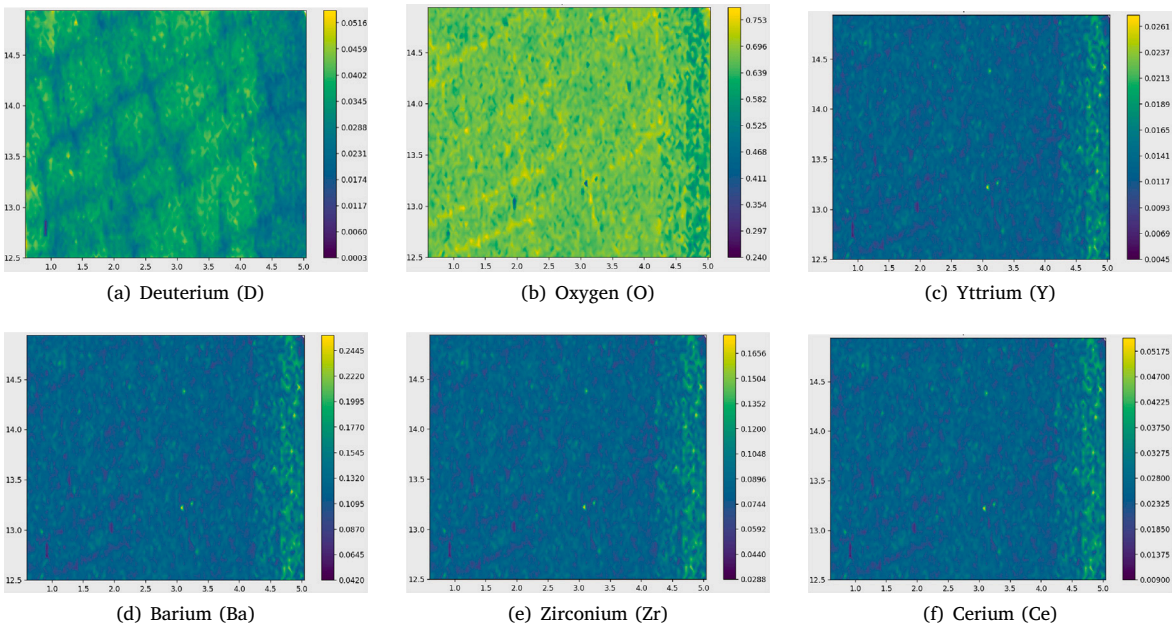
For an ideal model with Gaussian-distributed errors and known variances, the expected value of the minimum χ^2 equals the number of degrees of freedom. In the present work, the per-ROI objective function is defined using the Neyman χ^2 statistic [Eq. (4)], with variances estimated from the measured counts and regularized as $\max(M_i, 1)$. As a result, the absolute χ^2 values do not follow a strict χ^2 distribution and should be interpreted as a relative measure of fit quality rather than a probabilistic goodness-of-fit metric.

Within this framework, the higher χ^2 values observed for the RBS spectra are attributed to residual systematic uncertainties, including detector response, stopping powers, and cross sections, as well as the sharper spectral features and higher count rates typical of RBS. This is illustrated in Fig. 3 (left), where localized deviations between measured and simulated spectra are visible in the channel range 1030–1100.

In contrast, the lower χ^2 values obtained for the NRA spectra arise from the large fraction of low-count channels, for which the $\max(M_i, 1)$ regularization reduces the effective statistical weight of individual channels. A representative NRA fit is shown in Fig. 3 (right). The limited spatial variation of χ^2 across the scanned area indicates consistent fit quality and supports the robustness of the automated analysis workflow.

Table 1
Objective function minima from 31 BZCY721 spectrum fits (N_{max} = 1900). Lower values indicate better fits; the best result per row is highlighted in bold.

	LSHADE	JADE	ARRDE	GSA	BIPOP-CMAES	DMS-PSO	L-BFGS-B	NM
Mean	3461	3685	3428	3648	6379	3487	3645	**3424**
Median	3485	3641	3464	3591	5937	3501	3634	**3439**
Std	170	228	166	288	2467	**158**	255	169



(a) Deuterium (D) (b) Oxygen (O) (c) Yttrium (Y)

(d) Barium (Ba) (e) Zirconium (Zr) (f) Cerium (Ce)

Fig. 4. Heatmaps of elemental concentrations extracted at a depth of 1.5×10^{23} atoms cm^{-2} ($\approx$ 2 μm). The x and y axes indicate the spatial coordinates in millimeters.

Fig. 4 presents the elemental concentration maps of deuterium (D), oxygen (O), yttrium (Y), barium (Ba), zirconium (Zr), and cerium (Ce), extracted at an areal density of 1.5×10^{23} atoms m^{-2}, corresponding to a depth of approximately 2 μm. Barium, zirconium, cerium, and yttrium exhibit nearly identical spatial distributions and contrast levels, reflecting the fixed stoichiometric ratios imposed on their EDP parameters. In contrast, deuterium shows a distinctly different pattern, forming a grid-like structure with locally elevated concentrations. The lateral-averaged deuterium content ranges between 3.3% and 3.8% across the analyzed depth, but its spatial distribution is highly inhomogeneous: within the grid lines of the reconstructed deuteron profile (see Fig. 4(a)), the concentration drops below 0.1%, whereas in the centers of the grid cells it reaches values up to 8%.

Because deuterium serves as a tracer for heavy water (D_2O), its spatial presence directly reflects residual water retained within the material after cold sintering. During the cold-sintering process, water acts as a transient medium that promotes particle rearrangement and densification. Although most of this water is expected to evaporate during the final heating stage — typically leaving less than 0.1–0.5% residual content — the measured deuterium concentration of approximately 3% suggests that a significant fraction of heavy water remains trapped within the microstructure.

Thermogravimetric (TG) and differential thermal analysis (DTA) further support this observation, revealing a moisture loss of about 2.4% when the as-cold-sintered specimen, prepared using heavy water as the sintering aid, was heated from room temperature to 250 °C. The mass loss observed in TG/DTA closely corresponds to the residual D_2O content estimated from the IBA measurements. Nevertheless, part of the deuterium signal observed in the elemental maps may also arise from $Ba(OD)_2$, which can form as a by-product of the reaction between $BaCeO_3$ and D_2O.

The non-negligible presence of residual water is a significant finding, as it indicates incomplete removal of the transient sintering aid and reveals a potential source of mechanical degradation. Trapped water can generate localized vapor pressure during subsequent thermal processing or operation, potentially leading to microcrack formation and compromising the mechanical integrity of the BZCY721 layer. Consequently, the deuterium map provides direct evidence of retained heavy water and pinpoints specific regions where microstructural damage is most likely to initiate.

4. Conclusion

We have presented AutoNRA, a high-throughput framework for self-consistent ion beam analysis that integrates SIMNRA into a parallelized and automated pipeline. By exploiting multiprocessing, AutoNRA accelerates spectrum fitting by up to two orders of magnitude compared to conventional sequential workflows, enabling the analysis of thousands of spectra within a single experiment. The framework maintains consistent shared parameters across spectra from the same sample point while allowing independent evaluation across different points.

The capability of AutoNRA was demonstrated on a BZCY721 sample fabricated by cold sintering. The results revealed residual deuterium, indicating incomplete evaporation of heavy water used during sintering. This residual water content is of practical importance, as it may contribute to microcrack formation and reduced mechanical stability in the electrolyte layer.

Looking ahead, the modular architecture of AutoNRA readily accommodates the integration of additional IBA simulation engines beyond SIMNRA, such as future open-source or machine-learning-based simulators. This extensibility ensures that the framework can evolve in parallel with emerging computational models and experimental

techniques, ultimately serving as a unified platform for automated, high-throughput IBA analysis across diverse applications.

CRediT authorship contribution statement

S. Möller: Writing – review & editing, Validation, Resources, Data curation. **S.A. Afzal:** Writing – review & editing, Resources, Data curation. **M. Bram:** Writing – review & editing, Supervision. **M. Finsterbusch:** Writing – review & editing, Supervision, Funding acquisition.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Sören Möller reports a relationship with Aachen Ion Beams UG that includes: board membership and equity or stocks. If there are other authors, they declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This work was supported by the Federal Ministry of Education and Research (BMBF, Germany) under the project *For-Analytik* (grant no. 13XP0445).

Data availability

The AutoNRA software and the datasets analyzed in this study are available from the corresponding author upon reasonable request.

References

[1] S. Möller, Accelerator technology: applications in science, medicine, and industry, in: Particle Acceleration and Detection, Springer International Publishing, 2020.
[2] S. Möller, K.F. Muzakka, D. Höschen, M. Finsterbusch, High-throughput mev ion beam analysis - quantitative full stoichiometry imaging of a granite, Nucl. Instrum. Methods Phys. Res. Sect. B: Beam Interactions Mater. Atoms 560 (2025) 165626.
[3] Astrid Tazzioli, Quentin Lemasson, Alexandre Girard, Laurent Pichon, Brice Moignard, Claire Pacheco, Advanced 2d-pixe/rbs processing with machine learning at the new aglae facility for ancient layered objects, Nucl. Instrum. Methods Phys. Res. Sect. B: Beam Interactions Mater. Atoms 555 (2024) 165469.
[4] B.L. Doyle, P.P. Provencio, P.G. Kotula, A.J. Antolak, C.G. Ryan, J.L. Campbell, K. Barrett, Pixe-quantified axsia: Elemental mapping by multivariate spectral analysis, Nucl. Instruments Methods Phys. Res. Sect. B: Beam Interactions Mater. Atoms 249 (1) (2006) 828–832, Ion Beam Analysis.
[5] Khoirul Faiq Muzakka, Sören Möller, Stefan Kesselheim, Jan Ebert, Alina Bazarova, Helene Hoffmann, Sebastian Starke, Martin Finsterbusch, Analysis of rutherford backscattering spectra with cnn-gru mixture density network, Sci. Rep. 14 (1) (2024) 16983.
[6] Matej Mayer, Simnra User's Guide, 1997.
[7] T.F. Silva, C.L. Rodrigues, M. Mayer, M.V. Moro, G.F. Trindade, F.R. Aguirre, N. Added, M.A. Rizzutto, M.H. Tabacniks, Multisimnra: A computational tool for self-consistent ion beam analysis using simnra, Nucl. Instrum. Methods Phys. Res. Sect. B: Beam Interactions Mater. Atoms 371 (2016) 86–89, The 22nd International Conference on Ion Beam Analysis (IBA 2015).
[8] René Heller, Nico Klingner, Niels Claessens, Clement merckling, and johan meersschaut. differential evolution optimization of rutherford backscattering spectra, J. Appl. Phys. 132 (16) (2022) 165302.
[9] C. Jeynes, V.V. Palitsin, G.W. Grime, C. Pascual-Izarra, A. Taborda, M.A. Reis, N.P. Barradas, External beam total-iba using datafurnace, Nucl. Instrum. Methods Phys. Res. Sect. B: Beam Interactions Mater. Atoms 481 (2020) 47–61.
[10] ONNX Runtime developers, Onnx runtime, 2024, Version: 1.17. 0.
[11] Xiangpan Ji, Wenqiang Gu, Xin Qian, Hanyu Wei, Chao Zhang, Combined neyman–pearson chi-square: An improved approximation to the Poisson-likelihood chi-square, Nucl. Instrum. Methods Phys. Res. Sect. A: Accel. Spectrometers, Detect. Assoc. Equip. 961 (2020) 163677.
[12] Steve Baker, Robert D. Cousins, Clarification of the use of chi-square and likelihood functions in fits to histograms, Nucl. Instrum. Methods Phys. Res. 221 (2) (1984) 437–442.
[13] T.F. Silva, C.L. Rodrigues, M.H. Tabacniks, U. von Toussaint, M. Mayer, Bias and synergy in the self-consistent approach of data analysis of ion beam techniques, Nucl. Instrum. Methods Phys. Res. Sect. B: Beam Interactions Mater. Atoms 533 (2022) 9–16.
[14] Khoirul Faiq Muzakka, Sören Möller, Martin Finsterbusch, Minion : a high-performance derivative-free optimization library designed for solving complex optimization problems, 2025.
[15] Wenyu Zhou, Fanlin Zeng, Jürgen Malzbender, Hartmut Schlenz, Wendelin Deibert, Dmitry Sergeev, Ivan Povstugar, Ruth Schwaiger, Arian Nijmeijer, Michael M.üller, Olivier Guillon, Wilhelm Albert Meulenberg, Promoting densification and grain growth of $BaCe_{0.65}Zr_{0.2}Y_{0.15}O_{3-\delta}$, J. Mater. Res. Technol. 27 (2023) 3531–3538.
[16] Jianhua Tong, Daniel Clark, Michael Hoban, Ryan O'Hayre, Cost-effective solid-state reactive sintering method for high conductivity proton conducting yttrium-doped barium zirconium ceramics, Solid State Ion. 181 (11) (2010) 496–503.
[17] Clive A. Randall, Jing Guo, Amanda Baker, Michael Lanagan, Hanzheng Guo, Cold sintering ceramics and composites, 2017, Filed: 2016-09-27; Application number: US15/277, 553; Publication date: 2017-03-23.
[18] Sören Möller, Daniel Höschen, Sina Kurth, Gerwin Esser, Albert Hiller, Christian Scholtysik, Christian Dellen, Christian Linsmeier, A new high-throughput focused mev ion-beam analysis setup, Instruments 5 (1) 2021.
[19] Khoirul Faiq Muzakka, Ahsani Hafizhu Shali, Haris Suhendar, Sören Möller, Martin Finsterbusch, Robust differential evolution via nonlinear population size reduction and adaptive restart: The arrde algorithm, 2025, arXiv preprint arXiv: 2511.18429.
[20] Ryoji Tanabe, Alex S. Fukunaga, Improving the search performance of shade using linear population size reduction, in: 2014 IEEE Congress on Evolutionary Computation, CEC, 2014, pp. 1658–1665.
[21] Jingqiao Zhang, Arthur C. Sanderson, Jade: Adaptive differential evolution with optional external archive, IEEE Trans. Evol. Comput. 13 (5) (2009) 945–958.
[22] Nikolaus Hansen, Benchmarking a bi-population cma-es on the bbob-2009 function testbed. In, in: Proceedings of the 11th Annual Conference Companion on Genetic and Evolutionary Computation Conference: Late Breaking Papers, GECCO'09, Association for Computing Machinery, New York, NY, USA, 2009, pp. 2389–2396.
[23] J.J. Liang, P.N. Suganthan, Dynamic multi-swarm particle swarm optimizer., in: Proceedings 2005 IEEE Swarm Intelligence Symposium, 2005. SIS 2005., 2005, pp. 124–129.
[24] Y. Xiang, D.Y Sun, W. Fan, X.G Gong, Generalized simulated annealing algorithm and its application to the thomson model, Phys. Lett. A 233 (3) (1997) 216–220.
[25] Richard H. Byrd, Peihuang Lu, Jorge Nocedal, Ciyou Zhu, A limited memory algorithm for bound constrained optimization, SIAM J. Sci. Comput. 16 (5) (1995) 1190–1208.
[26] J.A. Nelder, R. Mead, A simplex method for function minimization, Comput. J. 7 (4) (1965) 308–313.