Real-Time Toroidal Equilibrium Reconstruction for RFX-mod2 via Quantized Neural Network in FPGA

L. Saccaro, *Graduate Student Member, IEEE*, A. Rigoni Garola, *Member, IEEE*, P. Zanca, D. Terranova, and R. Cavazzana

***Abstract*—RFX-mod2 will operate with an upgraded diagnostic and control system that can benefit from more accurate real-time plasma-state reconstruction. In particular, replacing the cylindrical approximation used in fast control applications with a toroidal equilibrium description is desirable, but the reference toroidal reconstruction routine is currently too computationally demanding for direct use in the real-time control cycle. This work presents a neural network (NN) surrogate for the toroidal equilibrium model in reversed field pinch operation, intended as the first block of a future real-time toroidal magnetic-perturbation reconstruction chain. The network approximates the parametric solution map from the radial coordinate and equilibrium parameters to the radial profiles of the toroidal and poloidal magnetic-field components and the Shafranov shift. A multi-objective architecture search is first used to identify compact floating-point models, followed by quantization-aware training (QAT) with heterogeneous precision assignment to reduce hardware cost while preserving accuracy. Selected quantized models are translated to field-programmable-gate-array (FPGA) firmware using hls4ml and synthesized and implemented for a Kria K26 platform. The final selected model preserves accuracy close to the floating-point baseline, reproduces derived quantities such as poloidal flux surfaces and rational-surface locations, and achieves an initiation interval (II) of one clock cycle. Post-implementation timing simulation of a wrapper-level design evaluating a 21-point radial grid indicates a full-scan latency of approximately 300 nanoseconds, well below the 100 microseconds period of the RFX-mod2 control cycle.**

***Index Terms*—Field programmable gate arrays, magnetic confinement, neural network hardware, neural networks, quantization (signal), real-time systems.**

I. Introduction

RFX-mod2 [1], [2] is the major upgrade of the RFX-mod experiment, conceived to extend its capabilities for the study and control of magnetically confined plasmas in multiple operating configurations. The new device introduces substantial modifications of the conducting structures surrounding the plasma: the ratio between shell radius and plasma minor radius has been reduced from $b/a = 1.10$ to 1.04, and the previous highly resistive vacuum vessel has been replaced by a low-resistivity copper shell. These changes are expected to mitigate intrinsic magnetic error fields and reduce the amplitude of magnetohydrodynamic (MHD) instabilities, thereby improving plasma performance and stability. A second major aspect of the upgrade concerns diagnostics and control infrastructure. RFX-mod2 is equipped with a significantly enhanced measurement system [3], designed to support advanced feedback control together with detailed investigations of plasma equilibrium, transport, and stability. In particular, the magnetic diagnostic includes approximately 1500 sensors with acquisition rates up to $1\,\mathrm{MHz}$ [4], [5], providing very high spatial and temporal resolutions. Building on the operational experience accumulated during RFX-mod campaigns (2004–2015), RFX-mod2 is currently in its final assembly phase and is expected to begin operation in the near future. As in its predecessor, the device is designed for flexible operation in tokamak, reversed field pinch (RFP), and ultra-low safety-factor plasma configurations.

The real-time control system developed for RFX-mod relies on a cylindrical approximation, adopted because full toroidal magnetic reconstruction is computationally too demanding for sub-millisecond feedback operation. While effective for fast control, this simplification neglects the true toroidal geometry of the device and therefore limits the fidelity of plasma state estimation. A toroidal reconstruction can provide a more accurate description of the equilibrium magnetic configuration, improve the estimation of magnetic perturbations, and better separate plasma-generated fields from externally applied control fields. These features are expected to enhance mode control and to more fully exploit the upgraded high-resolution diagnostic capabilities of RFX-mod2. This motivates the search for computationally efficient toroidal surrogate models.

Beyond this objective, this work is embedded in a broader long-term vision for plasma control in RFX-mod2 and other fusion experiments. The underlying concept is to combine heterogeneous diagnostics with physics-based priors in order to infer a compact and informative latent representation of the plasma state, suitable for real-time decision-making. Such an encoded plasma state could then be provided to advanced controllers, including model-based and, potentially, reinforcement-learning controllers, while maintaining compatibility with the existing control architecture.

This work addresses the first step in this direction, focusing on the development of a surrogate model for real-time toroidal equilibrium reconstruction. The main contributions are threefold:

1) We develop a neural network (NN) surrogate model that approximates the radial profiles of the poloidal and toroidal equilibrium magnetic-field components, together with the Shafranov shift.
2) We carry out a two-stage optimization procedure. A multi-objective architecture search first identifies the trade-off between reconstruction accuracy and model

Manuscript submitted for review on June 22, 2026.

Authors are with the Consorzio RFX (CNR, ENEA, INFN, Università di Padova, Acciaierie Venete SpA), Corso Stati Uniti 4, 35127 Padova, Italy

L. Saccaro (Graduate Student Member) is pursuing a Ph.D. in Fusion Science and Engineering at the Centro Ricerche Fusione — Università di Padova, 35131 Padova, Italy (email: lorenzo.saccaro@igi.cnr.it)

D. Terranova is with the Istituto per la Scienza e la Tecnologia dei Plasmi, CNR, Padova, Italy

size; then, quantization-aware training (QAT) converts the selected model from 32-bit floating-point arithmetic to fixed-point representations optimized for hardware deployment.

3) We deploy the optimized models on field-programmable gate array (FPGA) hardware using a high-level synthesis (HLS) workflow. The selected implementation achieves an initiation interval (II) of one clock cycle and a three-cycle single-point latency; post-implementation timing simulation of the wrapper-level design indicates a latency of approximately $300\,\mathrm{ns}$ for a 21-point radial scan, while preserving accuracy close to the single-precision floating-point (FP32) baseline.

The remainder of this paper is organized as follows. Section II introduces the RFP equilibrium model and reviews relevant NN and FPGA-based approaches for real-time plasma applications. Section III describes the generation of the reference dataset, the physics-constrained neural surrogate, the architecture search, the quantization-aware training procedure, and the hardware deployment workflow. Section IV presents the architecture-optimization, quantization, hardware-synthesis, and reconstruction-accuracy results. Section V discusses the implications of the results, the advantages and limitations of the proposed approach, and possible extensions. Section VI summarizes the main findings.

II. Scientific Background and State of the Art

A. Toroidal Equilibrium Model

The RFP is a toroidal magnetic confinement configuration in which the plasma is confined by both toroidal B_ϕ and poloidal B_θ magnetic field components. As in tokamaks, B_ϕ is directed along the torus and B_θ around the plasma column, but their relative strengths are markedly different. In tokamaks, the toroidal field is dominant and externally generated, while the poloidal field is weaker and mainly produced by the plasma current. In contrast, in the RFP the two components are of comparable magnitude, with a relatively strong poloidal field sustained by the large plasma current. This makes the RFP a high-current, low-field configuration. The defining feature of the RFP is the reversal of the toroidal field near the plasma edge: B_ϕ decreases radially and changes sign close to the boundary, becoming opposite to the core direction. Due to its low and strongly sheared safety factor profile, the RFP is prone to rich MHD activity, including tearing modes and self-organized states such as Multiple Helicity (MH) and Quasi Single Helicity (QSH), the latter generally associated with improved confinement [6].

The method proposed by [7] allows obtaining the radial profile of the magnetic perturbations in toroidal geometry. The model assumes a prescribed, force-free magnetic equilibrium ($\mathbf{J}\times\mathbf{B}=0, \nabla\times\mathbf{B}=\mu_0\mathbf{J}, \nabla\cdot\mathbf{B}=0$) in a toroidal configuration with circular cross-section, appropriate for low-β plasmas where pressure effects can be neglected. A perturbative approach is considered by dividing each quantity into a zeroth-order dominant axisymmetric component and a non-axisymmetric one $X(r,\theta,\phi)=X_0(r,\theta)+x(r,\theta,\phi)$. The equations are solved adopting a realistic toroidal geometry described using flux co-ordinates and an ideal shell as magnetic boundary, assuming that the zeroth-order flux surfaces have a circular cross-section and a differential radial shift. The shift with respect to the shell center is the function $\Delta(r)$. The zeroth-order current density $\mathbf{J}_0$ is found to be proportional to the magnetic field $\mathbf{B}_0$ through a function of r only: $\mu_0\mathbf{J}_0=\sigma(r)\mathbf{B}_0$, where $\sigma(r)$ is parametrized using the $\alpha-\Theta_0$ model $\sigma(r)=\frac{2\Theta_0}{a}\left(1-\frac{r}{a}\right)^\alpha$. Once the horizontal shift of the plasma surface Δ_H and the two parameters α and Θ_0 are specified, the zeroth-order radial profile of the poloidal and toroidal magnetic field, the radial profile of the Shafranov shift $\Delta(r)$ and the safety factor $q(r)$ can be computed. The α and Θ_0 parameters are determined from the pinch $\Theta(a)=\frac{B_\theta(a)}{\langle B_\phi\rangle}$ and the reversal $F(a)=\frac{B_\phi(a)}{\langle B_\phi\rangle}$ parameters, which are directly measured by the magnetic diagnostics. For compactness, we introduce the auxiliary functions:

$$\begin{aligned} f_\phi(r,\Delta,\Delta') &= \frac{\Delta}{R_0}+\frac{r}{2R_0}\Delta'-\frac{r^2}{2R_0^2},\\ f_\vartheta(r,\Delta,\Delta') &= \frac{r^2}{2R_0^2}+\frac{\Delta'^2}{2}+\frac{r}{2R_0}\Delta'-\frac{\Delta}{R_0}. \end{aligned} \tag{1}$$

Then, one has to solve the following system of ordinary differential equations (ODEs):

$$\begin{cases} \dfrac{d}{dr}\left[\dfrac{R_0}{r}\hat{B}_1^\phi\right] = -\sigma(r)\hat{B}_1^\vartheta,\\ \dfrac{d}{dr}\left[\dfrac{r}{R_0}\hat{B}_1^\vartheta\right] = \sigma(r)\hat{B}_1^\phi,\\ \Delta''+\dfrac{\Delta'}{r}\left(1+2r\dfrac{d\hat{B}_1^\vartheta/dr}{\hat{B}_1^\vartheta}\right)+\dfrac{1}{R_0}=0,\\ \dfrac{d}{dr}\left[\dfrac{R_0}{r}\hat{B}_2^\phi\right]+\dfrac{d}{dr}\left[\dfrac{R_0}{r}f_\phi\hat{B}_1^\phi\right] = -\sigma(r)\hat{B}_2^\vartheta,\\ \dfrac{d}{dr}\left[\dfrac{r}{R_0}\hat{B}_2^\vartheta\right]+\dfrac{d}{dr}\left[\dfrac{r}{R_0}f_\vartheta\hat{B}_1^\vartheta\right] = \sigma(r)\hat{B}_2^\phi. \end{cases} \tag{2}$$

Where R_0 is the major radius ($2.0\,\mathrm{m}$ for RFX-mod2), $\hat{B}_1^\phi$ and $\hat{B}_1^\vartheta$ are the first-order contravariant toroidal and poloidal magnetic field components, while $\hat{B}_2^\phi$ and $\hat{B}_2^\vartheta$ are second-order corrections in $\epsilon=b/R_0$, which are typically very small. From these quantities, combining with the measured plasma current I_p, the physical toroidal and poloidal equilibrium magnetic field components can be computed together with the safety factor.

B. Neural Networks for Plasma Modeling and Control

Artificial NNs are well-established universal function approximators, capable of learning complex nonlinear mappings directly from data while retaining very fast inference once trained. This makes them attractive surrogate models whenever repeated evaluations of computationally expensive physics models are required, or when real-time estimates must be produced from diagnostic measurements.

In fusion research, NNs have been successfully applied to plasma-state estimation, disruption prediction, surrogate

modelling, and advanced control. Early works already demonstrated real-time extraction of equilibrium parameters in tokamaks using NN mappings [8]. More recently, deep learning has shown strong predictive capability for disruptive instabilities in magnetically confined plasmas [9]. Deep generative models have been applied to diagnostic data reconstruction in RFX-mod, where a modified variational autoencoder with stochastic input masking and β-annealing was shown to outperform conventional B-spline interpolation for incomplete soft-x-ray temperature profiles, especially under moderate to high levels of missing data [10]. Neural-network surrogates have further been developed to accelerate computationally intensive transport models by several orders of magnitude, enabling their integration into real-time-oriented workflows [11]. In parallel, reinforcement learning has been successfully demonstrated for closed-loop plasma control, including shape and current control experiments on the TCV tokamak [12].

Building on these demonstrated capabilities, this work combines NN surrogate modelling with real-time implementation requirements, approximating a physics-based toroidal equilibrium solver with a compact architecture suitable for low-latency deployment.

C. Deployment of Neural Networks on Programmable Logic

Real-time plasma control requires inference engines with deterministic latency, high throughput, and limited power consumption. FPGAs are well suited to this role because NN operations can be implemented as deeply pipelined, massively parallel fixed-point circuits. This allows inference to be performed with predictable timing, avoiding the latency variability typically associated with general-purpose processors.

Low-latency NN deployment on FPGAs has been extensively developed in the high-energy physics community, where trigger systems impose sub-microsecond timing constraints. The `hls4ml` framework was introduced to translate trained NNs into HLS firmware, enabling FPGA inference with latencies on the order of hundreds of nanoseconds [13]. Subsequent developments extended this approach to convolutional and recurrent architectures, demonstrating that compression techniques such as pruning and quantization-aware training can strongly reduce FPGA resource usage while preserving accuracy [14], [15].

Quantization is therefore a key element of FPGA-oriented NN design. Libraries such as `QKeras` enable quantization-aware training of Keras models and their subsequent conversion to hardware through `hls4ml` [16]. More recently, High Granularity Quantization (HGQ) introduced gradient-based mixed-precision optimization at finer granularity, allowing the bit-width of weights, biases, and activations to be optimized automatically. This approach can achieve substantial reductions in latency and resource usage compared with more conventional fixed-precision or layer-wise quantization strategies [17].

Although FPGA-based NN inference is still less common in fusion than in particle physics trigger systems, recent works have demonstrated its relevance for real-time plasma diagnostics and control. A closely related example in RFX-mod used deep variational autoencoders to integrate soft-x-ray temperature information with magnetic-state data, exploiting quantization of the neural transfer functions to adapt the inference model to embedded programmable logic for real-time plasma analysis [18]. More recently, low-latency FPGA deployment has also been explored for tokamak MHD mode tracking, showing the suitability of this approach for feedback-oriented fusion applications [19].

In this work, these developments are combined to obtain a compact, quantized neural surrogate of the toroidal equilibrium model, suitable for deterministic low-latency execution within the real-time control chain.

III. Methods

A. Reference Dataset Generation

Using the reference toroidal equilibrium solver, we build a database of equilibrium profiles covering the range of input parameters ($\alpha \in [3.0, 10.0], \Theta_0 \in [1.3, 1.6], \Delta_H \in [-0.01, 0.01]$m) typically expected in RFX-mod2 operation. This range also includes pre-reversal configurations, commonly observed during the initial phase of the discharge before edge field reversal is established, making equilibrium reconstruction in this phase relevant for control-oriented applications.

More specifically, the datasets are generated as follows:

- **Training set:** the parameter space $(\alpha, \Theta_0, \Delta_H)$ is discretized on a regular grid of $101 \times 101 \times 11 = 112{,}211$ operating points. For each point, the equilibrium profiles are evaluated on 100 equally spaced radial locations in the interval $r \in [r_0, b]$, where $r_0 > 0$ is a small positive offset introduced to avoid the singularity at the origin.
- **Validation set:** 100,000 parameter combinations are sampled uniformly over the same operating range. For each sample, 250 radial locations are uniformly drawn in the interval $r \in [r_0, b]$.

The resulting training and validation sets contain approximately 1.1×10^7 and 2.5×10^7 input—target pairs, respectively. Each input sample is given by

$$(r, \alpha, \Theta_0, \Delta_H),$$

while the corresponding targets are

$$\left(\hat{B}_1^{\phi}(r), \hat{B}_1^{\vartheta}(r), \Delta(r), \Delta'(r), \hat{B}_2^{\phi}(r), \hat{B}_2^{\vartheta}(r)\right).$$

Inputs are scaled to the $[0, 1]$ range. This normalization improves training stability and also facilitates the subsequent quantization step, since unsigned fixed-point formats can be used, saving the sign bit and limiting the required integer bit-width.

B. Physics-Constrained Neural Surrogate

Rather than predicting equilibrium profiles through an unconstrained regression model, the proposed network is designed to approximate the parametric solution map of the ODE system while satisfying the prescribed initial conditions by construction. This formulation reduces function complexity near the singular core region and improves learning across the

parametric operating domain, as the network can focus on the non-trivial part of the flow map.

Formally, if we denote with $\mathbf{y} = [\hat{B}_1^{\phi}, \hat{B}_1^{\vartheta}, \Delta, \Delta', \hat{B}_2^{\phi}, \hat{B}_2^{\vartheta}]$ the vector of target quantities, and with $\boldsymbol{\lambda} = [\alpha, \Theta_0, \Delta_H]$ the vector of input parameters, the ODE system in (2) can be rewritten as:

$$\frac{d\mathbf{y}}{dr} = \mathbf{f}(\mathbf{y}, r, \boldsymbol{\lambda}), \qquad \mathbf{y}(r_0) = \mathbf{y}_0(\boldsymbol{\lambda}). \tag{3}$$

The parametric flow map of the system can be defined as:

$$\boldsymbol{\Phi}_{r,r_0}(\mathbf{y}_0, \boldsymbol{\lambda}) = \mathbf{y}(r, \boldsymbol{\lambda}) \tag{4}$$

$$\mathbf{y}_{\mathrm{NN}}(r; \boldsymbol{\lambda}) = \mathbf{y}_0(\boldsymbol{\lambda}) + (r - r_0)\mathbf{N}_{\theta}(r, \boldsymbol{\lambda}), \tag{5}$$

where $\mathbf{N}_{\theta}$ is the NN with learnable parameters θ, approximating the non-trivial part of the flow map. Therefore, the learning problem can be formulated as the following optimization problem:

$$\min_{\theta} \mathbb{E}_{r,\boldsymbol{\lambda}} \left[\mathcal{L}(\mathbf{y}_{\mathrm{NN}}(r; \boldsymbol{\lambda}), \boldsymbol{\Phi}_{r,r_0}(\mathbf{y}_0, \boldsymbol{\lambda})) \right] \tag{6}$$

where $\mathcal{L}$ is a suitable loss function.

C. Architecture Design and Optimization

The base architecture consists of a multi-layer perceptron (MLP) with fully connected layers and rectified linear unit (ReLU) activation functions. A linear activation is adopted for the output layer in order to allow unconstrained predictions over the target variables. This design choice is motivated by the need to keep the network as simple and hardware-efficient as possible for the subsequent quantization and FPGA deployment. In particular, ReLU activations are well suited to digital implementations, as they can be realized through simple thresholding operations without requiring costly non-linear functions, usually implemented via look-up tables.

The model is implemented using Keras 3 [20], adopting the TensorFlow backend [21]. The use of Keras provides a backend-agnostic development framework and facilitates interoperability with downstream quantization and hardware-conversion toolchains. Hyperparameter optimization is carried out with Optuna [22], an open source hyperparameter optimization framework, using the Non-dominated Sorting Genetic Algorithm II (NSGA-II) sampler [23], which is well suited for multi-objective search problems. A total of 1000 optimization trials are performed, exploring a search space that includes the number of layers (between 1 and 5) and the number of neurons per layer (between 10 and 500).

The optimization simultaneously minimizes two competing objectives: the validation loss and the total number of trainable parameters. For each trial, candidate architectures are trained using the Adam optimizer with mean squared error (MSE) loss. The learning rate is adaptively reduced when the validation loss reaches a plateau, while early stopping is employed to terminate training once no significant improvement is observed.

Candidate architectures are selected from the Pareto-optimal set and forwarded to the subsequent quantization-aware training stage.

TABLE I
RESOLUTION OF THE NEURAL NETWORK INPUTS AFTER QUANTIZATION.

Input	Range	Resolution
r	$[1.07 \times 10^{-4}, 0.537]$ m	3.28×10^{-5} m
α	$[3.0, 10.0]$	4.27×10^{-4}
Θ_0	$[1.3, 1.6]$	1.83×10^{-5}
Δ_H	$[-0.01, 0.01]$ m	1.22×10^{-6} m

D. Quantization-Aware Training

To enable efficient FPGA deployment, the selected floating-point architecture is re-implemented using the HGQ2 library [17], replacing standard dense layers with their quantized version. This framework supports heterogeneous quantization, allowing independent precision assignment across weights, biases, and activations, which are optimized through a gradient-based approach. Pruning naturally emerges as the limiting case of zero-bit precision.

The initial bit-width for internal weights, biases, and activations is set to approximately 10 bits. In contrast, the input and output formats are fixed a priori and excluded from optimization in order to satisfy interface and numerical-accuracy constraints. Inputs are represented using unsigned 15-bit fixed-point numbers with one integer bit, while outputs use signed 24-bit fixed-point representation composed of one sign bit, two integer bits, and 21 fractional bits. For the inputs, after mapping the values back to the original range, the corresponding resolutions are reported in TABLE I, while for the outputs the resolution is approximately 4.8×10^{-7} for all target variables.

The training objective is defined as a weighted sum of the MSE loss and a hardware-cost regularization term based on Effective Bit Operations (EBOPs). EBOPs provide a proxy metric for silicon usage, with an upper limit given by $\approx 55 \times \mathrm{DSP} + \mathrm{LUT}$, where DSP and LUT denote digital signal processing slices and look-up tables, respectively, when using `hls4ml` for HLS conversion. The trade-off between accuracy and resource cost is controlled by the coefficient β:

$$\mathcal{L} = \mathrm{MSE} + \beta\,\mathrm{EBOPs}, \tag{7}$$

Starting from the pretrained floating-point model, QAT is performed in two stages. During an initial adaptation phase, the network is trained with quantization enabled while minimizing only the reconstruction loss, allowing the parameters to adjust smoothly to discretized arithmetic. The coefficient β is then introduced through a scheduling strategy: it is first raised to a prescribed initial value and then gradually decreased during training, encouraging early resource minimization while allowing later recovery of predictive accuracy. Several scheduling profiles were evaluated to explore different regions of the accuracy—resource trade-off space, by changing the initial value of β and the decay rate. The optimizer learning rate follows a cosine-annealing schedule with warm restarts. Unlike the floating-point architecture search stage, no early stopping is applied, and each quantized model is trained for a fixed number of epochs to ensure consistent convergence behavior. Throughout QAT, intermediate checkpoints

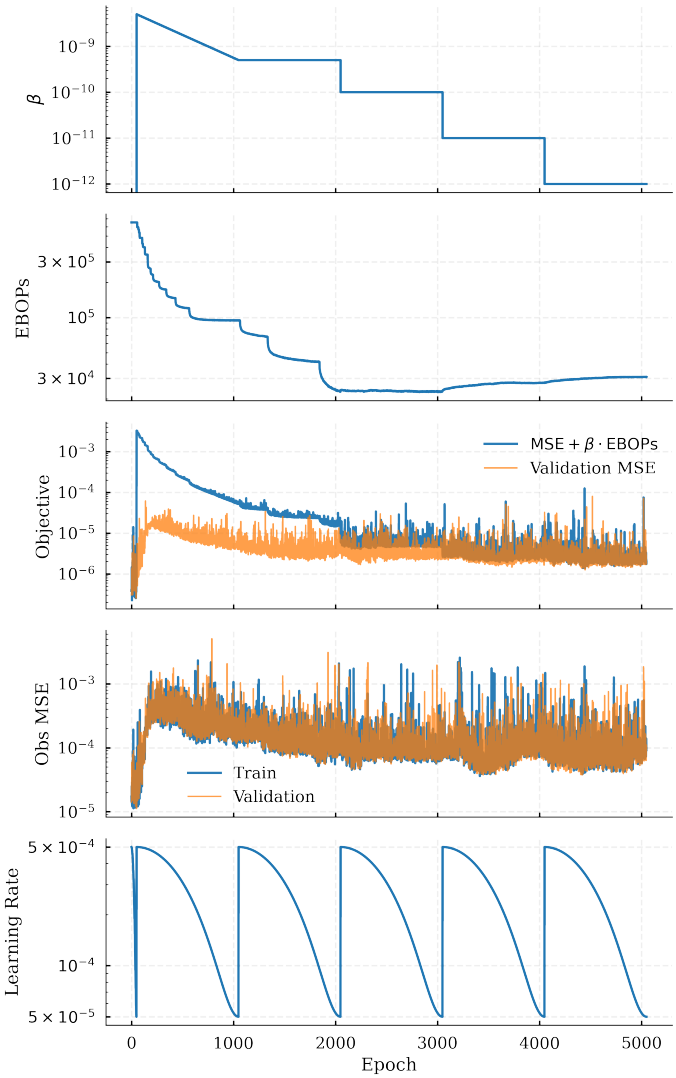

Fig. 1. Example training history during QAT, showing reconstruction loss, observed MSE, and EBOP regularization dynamics.

are ranked according to validation MSE and EBOPs. Only non-dominated solutions are retained, yielding a Pareto set of quantized models for subsequent HLS synthesis and hardware evaluation. An example of the training curves for a single quantization configuration is shown in Fig. 1. Here, Obs. MSE denotes the mean squared error computed on the reconstructed physical quantities (B_ϕ, B_θ, q).

E. Hardware Deployment Workflow

The quantized NN is translated into synthesizable HLS code using `hls4ml` version 1.3.0 [24], which provides an automated workflow for converting trained machine-learning models into FPGA-ready firmware. The conversion is configured to preserve bit-exact consistency with the HGQ2 quantized model, ensuring that hardware inference reproduces the numerically quantized software predictions.

In this study, priority is given to minimum achievable inference latency. To this end, a clock period of 10 ns is specified, the input/output (I/O) is set to `io_parallel`, where data are passed in parallel between the layers, and the overall implementation strategy is set to `Latency`. This configuration prioritizes full parallelism and aggressive pipelining, allowing the shortest possible execution time compatible with the available device resources.

HLS synthesis and subsequent implementation are carried out using Vivado 2024.1 [25]. Final deployment targets the AMD Kria K260 Robotics Starter Kit [26], based on the Zynq UltraScale+ Multiprocessor System-on-Chip (MPSoC) platform. This device was selected because it offers a favorable balance between cost and available FPGA resources, while also corresponding to an established in-house platform with prior experience on the Zynq UltraScale+ architecture.

The adopted workflow enables rapid transition from trained software models to deployable low-latency firmware, providing direct estimates of latency, throughput, and hardware utilization metrics such as LUTs, flip-flops (FFs), and DSP slices. This workflow provides the latency, throughput, and resource metrics used to compare candidate quantized models on the target device.

IV. Results

A. Architecture Search and Floating-Point Baseline

The results of the multi-objective architecture search are reported in Fig. 2, where validation MSE is shown as a function of the number of trainable parameters. Each point corresponds to one Optuna trial, while highlighted markers denote Pareto-optimal solutions.

A clear trend emerges: increasing the model size leads to substantial accuracy improvements in the low-parameter regime, with the validation error decreasing by more than two orders of magnitude between 10^2 and approximately 5×10^3 parameters. Beyond this point, the Pareto front enters a plateau region, where additional parameters provide only marginal gains. For networks larger than approximately 5×10^4 parameters, further improvements are gradual despite a one-order-of-magnitude increase in model size.

This behavior indicates that relatively compact architectures are sufficient to capture the dominant nonlinear structure of the equilibrium mapping, while larger models mainly offer diminishing returns. Consequently, model selection was focused on the knee region of the Pareto front, where the best trade-off between predictive accuracy and implementation cost is achieved.

Based on this criterion, two representative one-hidden-layer architectures were retained for subsequent quantization and hardware evaluation: a compact model with 280 hidden units (approximately 3×10^3 parameters), located near the onset of the plateau, and a larger model with 490 hidden units (approximately 5×10^3 parameters), corresponding to the last significant improvement before saturation.

Interestingly, the Pareto-optimal solutions in this region are consistently dominated by shallow networks with a single wide hidden layer, suggesting that the target mapping benefits more from width than from depth within the explored search space.

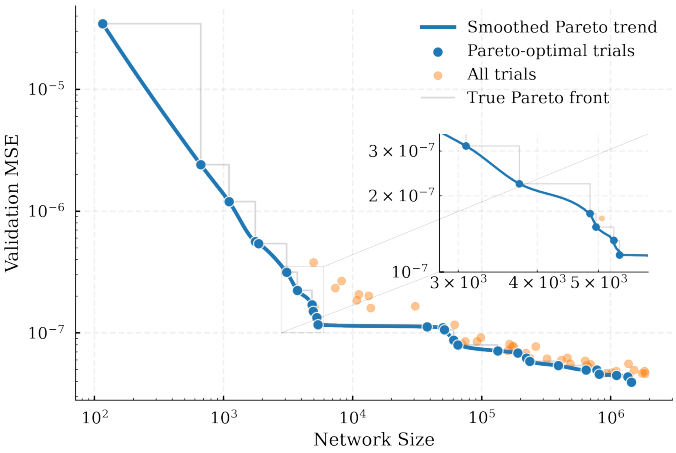

Fig. 2. Pareto front obtained from the architecture search. Validation MSE is reported as a function of network size (number of trainable parameters). Orange markers indicate all explored trials, blue markers Pareto-optimal solutions, and the solid curve shows a smoothed Pareto trend. The inset highlights the knee region used for selecting candidate architectures for quantization.

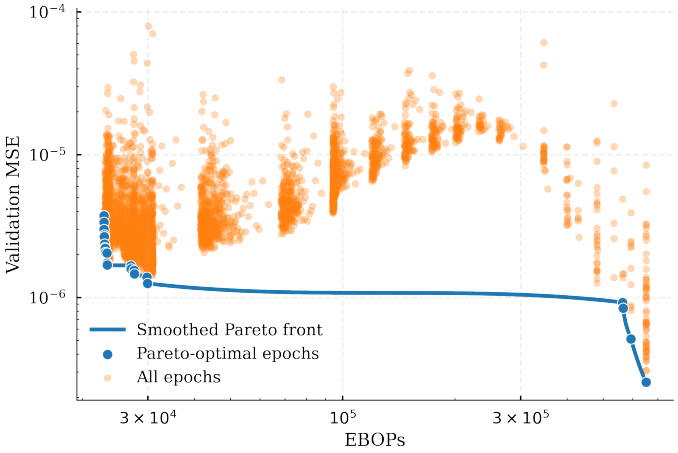

Fig. 3. Validation MSE versus EBOPs for intermediate checkpoints during a representative QAT run. Only non-dominated solutions are retained along training.

B. Quantization-Aware Optimization Results

The behavior of the proposed QAT strategy is illustrated in Fig. 3, where intermediate checkpoints from a representative training run are projected in the validation MSE—EBOPs plane. As described in Section III-D, training is initialized from the pretrained floating-point model and proceeds in two stages: an adaptation phase minimizing reconstruction loss only, followed by activation of the hardware-cost regularization term.

The resulting trajectory shows that substantial reductions in EBOPs can be achieved while preserving nearly constant validation error over a broad region of the Pareto front. Only in the most aggressive compression regime does the validation loss increase significantly, indicating that moderate quantization can remove a large fraction of arithmetic cost with limited impact on predictive accuracy.

To explore different operating points, several β-scheduling strategies were evaluated by varying both the initial regularization strength and its decay rate. The combined Pareto fronts are reported in Fig. 4. Conservative schedules favor accuracy-oriented solutions, while more aggressive schedules shift the frontier toward lower EBOP values at the cost of increased reconstruction error. Notably, the 280-unit architecture trained with a conservative configuration approaches the floating-point baseline accuracy while still providing a substantial reduction in effective bit operations (approximately a twofold reduction with respect to the initial homogeneous bit-width configuration), confirming that a certain amount of quantization can be introduced with negligible performance degradation. For all but the most aggressive β-scheduling policies, the Pareto fronts retain the expected capacity ordering: the wider model provides lower reconstruction error, whereas the smaller model achieves reduced EBOPs. Under stronger regularization, this monotonic trend is no longer guaranteed, indicating that scheduler dynamics can outweigh nominal model size.

The heterogeneous precisions learned for the 280-unit model with the most aggressive schedule are shown in Fig. 5. Several relevant trends emerge. First, pruning is automatically recovered in multiple parameters through zero-bit assignments. Second, the internal hidden-layer weight matrices are quantized to very low precision for most coefficients, with only a limited number of connections retaining higher bit-width. Third, the output projection layer systematically receives higher precision, indicating that most numerical sensitivity is concentrated in the final reconstruction stage.

These observations suggest that the equilibrium mapping admits a highly compressed internal representation, while accurate recovery of the physical target variables primarily requires precision in the final linear combination.

Candidate models for HLS conversion were selected as the lowest-validation-MSE solutions within the region where the achieved EBOPs were significantly lower than the corresponding starting point. In this way, hardware synthesis was focused on models providing meaningful computational savings without sacrificing predictive accuracy more than necessary. The selected candidates are summarized in TABLE II, together with their corresponding FP32 baselines. The table reports the validation MSE on the normalized network targets, the observed MSE on the reconstructed physical quantities (B_ϕ, B_θ, q), and the EBOPs used as hardware-cost proxy during QAT.

TABLE II
VALIDATION ACCURACY AND EBOPS OF THE SELECTED MODELS.

Model	EBOPs ($\times 10^3$)	Val. MSE	Val. Obs. MSE
490 FP32	–	1.17×10^{-7}	1.51×10^{-5}
490 conservative	262.9	2.69×10^{-7}	1.05×10^{-5}
490 moderate	141.4	4.63×10^{-7}	2.09×10^{-5}
490 aggressive	30.0	1.23×10^{-6}	5.25×10^{-5}
280 FP32	–	3.04×10^{-7}	2.29×10^{-5}
280 conservative	167.4	3.27×10^{-7}	2.57×10^{-5}
280 moderate	96.6	6.38×10^{-7}	3.36×10^{-5}
280 aggressive	29.8	9.78×10^{-7}	8.15×10^{-5}

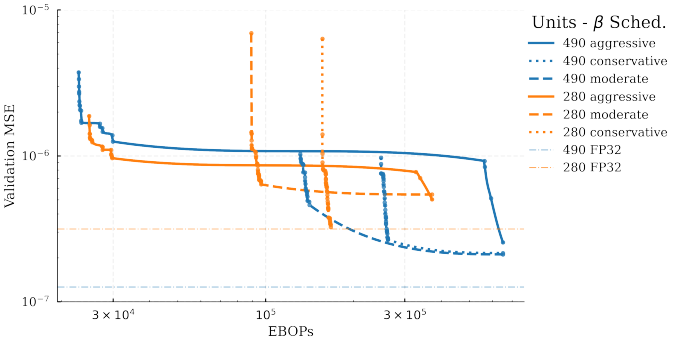

Fig. 4. Combined Pareto fronts obtained using different β-scheduling strategies for the 280- and 490-unit architectures.

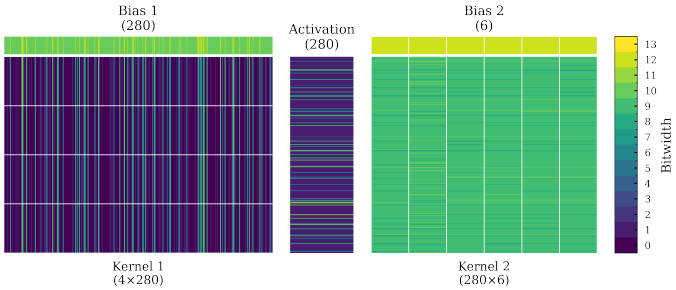

Fig. 5. Learned heterogeneous bit-width allocation for the aggressively compressed 280-unit model. Zero-bit entries correspond to pruned parameters.

C. Hardware Synthesis and Resource Evaluation

Table III reports the HLS C-synthesis estimates and the post-place resource utilization for the selected quantized models. All configurations achieve an II of one clock cycle, while the latency of the NN inference kernel ranges between two and three clock cycles, corresponding to 20–30 ns at the target 10 ns clock period.

A notable difference is observed between the C-synthesis estimates and the post-place resource utilization. In particular, the final LUT usage after Vivado implementation is substantially lower than the HLS C-synthesis estimate for all configurations, with reductions larger than a factor of two. For example, the 280-unit aggressive model decreases from 25460 LUTs at C-synthesis to 7323 LUTs after placement, while the 490-unit conservative model decreases from 88329 to 26419 LUTs. This indicates that the C-synthesis report provides a conservative estimate of LUT utilization and that subsequent logic optimization during implementation significantly reduces the final fabric footprint.

Conversely, the post-place DSP utilization is higher than the C-synthesis estimate. This increase is likely due to technology mapping and timing-driven implementation choices, where part of the arithmetic is mapped to DSP blocks to meet the target clock period. The effect is more pronounced for the moderate and conservative configurations, where the learned quantization is less aggressive and several weights retain higher bit-widths. In these cases, multiplications above approximately 7–8 bits are more likely to be implemented using DSP resources rather than LUT-based logic, depending on the synthesis and mapping strategy adopted by the Vivado flow.

Overall, the aggressive configurations provide the lowest LUT utilization, while the conservative configurations retain higher numerical precision at the cost of increased DSP and LUT usage. This confirms the expected accuracy—resource trade-off induced by the QAT procedure.

D. Reconstruction Accuracy of the Selected Model

Among the synthesized candidates, the 280-conservative configuration was selected for detailed analysis, as it provides the best compromise between reconstruction accuracy and hardware cost. In particular, it preserves an accuracy close to that of the corresponding FP32 baseline while requiring only moderate FPGA resources, making it a suitable choice for integration into the real-time control chain.

To assess the physical reconstruction quality, the selected quantized model is compared with both the reference numerical solver and the corresponding floating-point network for a representative RFP equilibrium extracted from an RFX-mod discharge.

Fig. 6 compares the radial profiles predicted by the selected quantized model and by the corresponding FP32 model with the reference numerical solution. Overall, the agreement is very good for the dominant quantities, including the first-order toroidal and poloidal magnetic-field components and the Shafranov shift terms. The quantized model closely follows the floating-point baseline and remains nearly indistinguishable from the numerical solution over most of the radial domain. The main visible discrepancy appears in the reconstruction of the second-order correction $\hat{B}_2^{\phi}$. However, this term is intrinsically much smaller than the leading-order equilibrium quantities, so that a larger relative deviation does not translate into a significant impact on the reconstructed physical equilibrium. The relative-error profiles confirm this interpretation. Over most of the radial domain, the relative error remains close to or below the 1% threshold. Larger relative deviations occur mainly where the reference quantities approach zero, in particular near the plasma edge, where the toroidal magnetic field and, consequently, the safety factor first approach zero and then change sign. Because relative error can become artificially large when the reference quantity approaches zero, its interpretation must be complemented by the corresponding profile magnitude.

The impact of these profile-level discrepancies on the reconstructed equilibrium is further assessed in Fig. 7, where the poloidal flux $\Psi(R, Z)$ reconstructed from the numerical solver, the FP32 model, and the selected quantized model is compared in the poloidal cross-section. The three flux maps are visually almost indistinguishable, indicating that the neural surrogate preserves the overall equilibrium geometry with high fidelity. In particular, the shape and spacing of the flux surfaces are accurately reproduced, showing that the approximation errors accumulated at the profile level do not significantly distort the reconstructed equilibrium. The corresponding error maps confirm that the discrepancy remains limited over the whole cross-section. The largest deviations are localized in the core region, while most of the plasma volume exhibits very small errors.

TABLE III
HLS C-SYNTHESIS ESTIMATES AND POST-PLACE RESOURCE UTILIZATION FOR SELECTED QUANTIZED MODELS FOR THE KRIA K26.

Model	EBOPs	Latency	II	C-Synthesis Estimate			Post-Place Utilization		
	($\times 10^3$)	cycles / ns	cycles	DSP	FF	LUT	DSP	FF	LUT
490 aggressive	30.0	3 / 30	1	105 (8.4%)	2004 (0.9%)	35997 (30.7%)	203 (16.3%)	1569 (0.7%)	9374 (8.0%)
490 moderate	141.4	2 / 20	1	155 (12.4%)	864 (0.4%)	49692 (42.4%)	222 (17.8%)	869 (0.4%)	15048 (12.8%)
490 conservative	262.9	3 / 30	1	265 (21.2%)	3036 (1.3%)	88329 (75.4%)	367 (29.4%)	2626 (1.1%)	26419 (22.6%)
280 aggressive	29.8	3 / 30	1	113 (9.1%)	1344 (0.6%)	25460 (21.7%)	166 (13.3%)	908 (0.4%)	7323 (6.3%)
280 moderate	96.6	2 / 20	1	121 (9.7%)	1026 (0.4%)	34657 (29.6%)	168 (13.5%)	891 (0.4%)	11461 (9.8%)
280 conservative	167.4	3 / 30	1	188 (15.1%)	2332 (1.0%)	65316 (55.8%)	271 (21.7%)	2022 (0.9%)	19753 (16.9%)

Finally, the reconstruction accuracy was evaluated on the radial position of a few relevant resonant surfaces, defined by the condition $q = m/n$, where m and n are the poloidal and toroidal mode numbers, respectively. In particular, the error in the reconstructed surface radius was computed over a subset of 500 RFP equilibria for the reversal surface $q = 0$, associated with $m = 0$ modes, and for the $m = 1$ rational surfaces with $n = 7, 8, 9$.

The resulting error distributions are reported in Fig. 8. The selected quantized model retains a localization accuracy comparable to the FP32 baseline for the dominant rational surfaces, with typical errors on the order of a few millimeters. For most surfaces and configurations, the typical error remains on the order of 1–$2\,\mathrm{mm}$, corresponding to only a few tenths of a percent of the plasma minor radius. As expected, larger spreads are observed for rational surfaces located in regions where the safety-factor profile is flatter, as in the case of $q = 1/7$, since small errors in $q(r)$ can produce larger shifts in the inferred crossing position. This behavior affects both the FP32 and quantized models, with similar trends in the error distributions.

The comparison across quantization schedules is consistent with the validation metrics reported above: conservative configurations provide the most accurate localization, while aggressive configurations increase the spread of the error distribution in exchange for lower hardware cost. This result is particularly relevant from the control perspective and for the subsequent computation of perturbation profiles, since it shows that the selected quantized model not only predicts the underlying equilibrium variables accurately, but also preserves derived quantities of direct physical interest.

V. DISCUSSION

The results obtained in this work suggest that the proposed learning problem is well suited to a NN surrogate approach. The modelling choice deliberately limits the NN task to the approximation of a well-defined parametric solution map. The input space is low dimensional, consisting only of the radial coordinate and the equilibrium parameters α, Θ_0, and Δ_H, and the relevant operating ranges are known a priori from the expected RFX-mod2 scenarios. This makes it possible to sample the parameter space systematically with a dense grid and to train the network primarily as an interpolator within the physically relevant domain, rather than as a model required to extrapolate from sparse or poorly constrained data. As a consequence, the generalization burden is reduced, limiting the need for additional regularization techniques such as dropout [27] or explicit L2 penalties.

It should be emphasized that the present validation assesses the surrogate against the reference toroidal equilibrium solver over the prescribed RFX-mod2 operating domain; validation against future experimental RFX-mod2 discharges and integration with the complete perturbation-reconstruction chain remain subjects of future work.

A possible improvement concerns the smoothness and differential consistency of the reconstructed profiles. In the present implementation, the network is trained by minimizing pointwise reconstruction errors with respect to the numerical solution. Future developments could include physics-informed loss terms, similarly to physics-informed neural networks (PINNs) [28], by exploiting automatic differentiation to compute derivatives with respect to the radial coordinate r. This would allow the residual of the governing ODE system to be directly penalized during training, encouraging smoother profiles and improving consistency between the predicted quantities and their radial derivatives.

The quantization results confirm the advantage of heterogeneous precision assignment for FPGA-oriented neural surrogates. Instead of imposing a uniform bit-width across the full model, HGQ allows the training process to identify which parameters, biases, and activations require higher numerical precision and which can be aggressively quantized or pruned. The learned bit-width distributions show that most internal weights can be represented with very low precision, while higher precision is mainly retained in the output layer. This behavior is consistent with the role of the hidden layer as a compressed internal representation and of the final projection as the stage where the physical target quantities are reconstructed.

At the same time, the QAT phase introduces an additional optimization problem. Compared with the floating-point architecture search, quantized training requires longer training times and a careful choice of the β-scheduler controlling the EBOP regularization term. Different schedules explore different regions of the validation-MSE—EBOPs trade-off, and the final hardware efficiency depends not only on the nominal model size, but also on how the precision allocation evolves during training. For this reason, architecture optimization and quantization should not be treated as fully independent steps. Although the present work applies them sequentially, future work could combine the two stages into a single hardware-aware optimization loop, jointly selecting

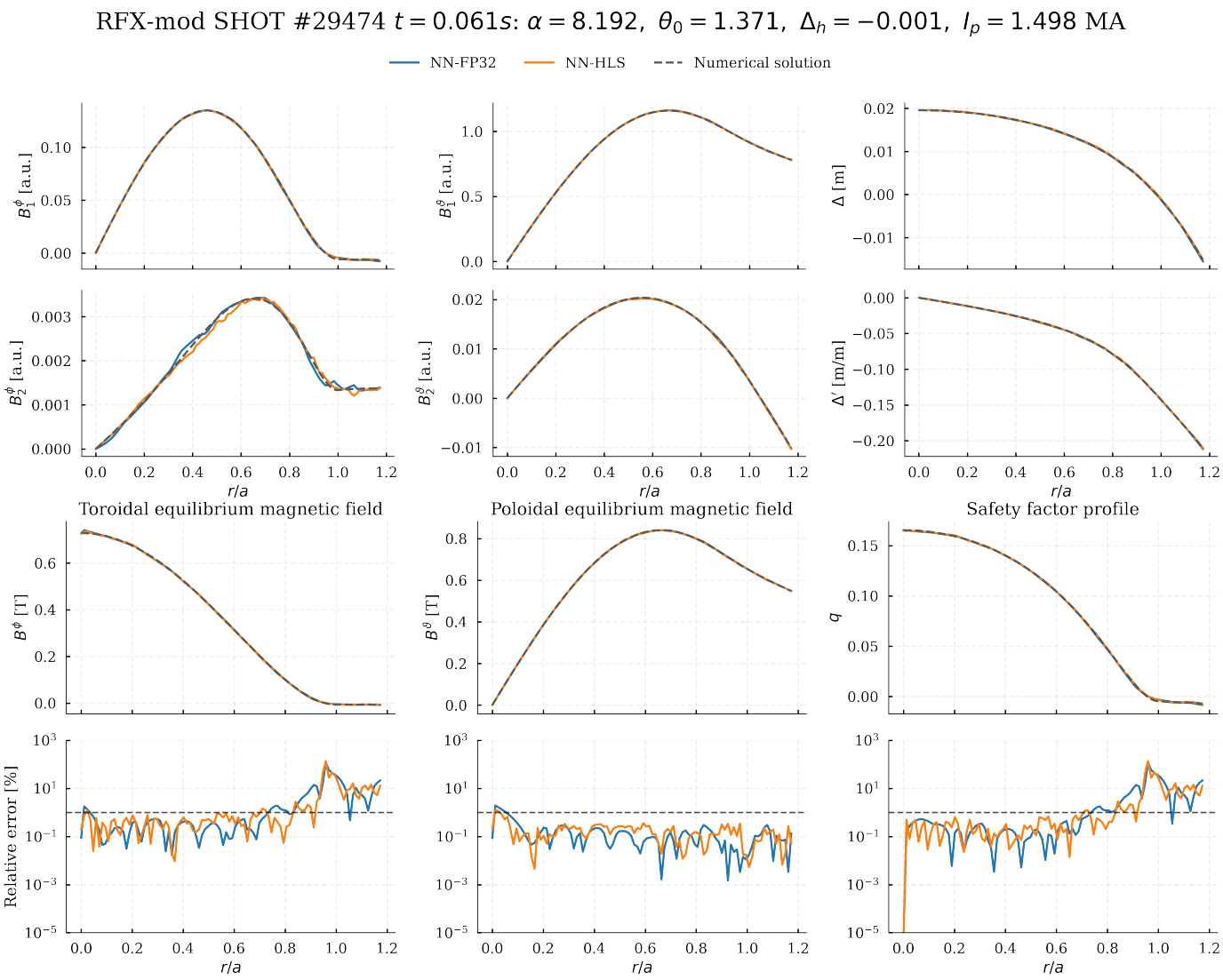

Fig. 6. Comparison of reconstructed radial profiles for a representative RFP equilibrium extracted from an RFX-mod discharge. The numerical solution, the FP32 surrogate, and the selected quantized model are shown for the target quantities, the corresponding physical equilibrium quantities, and their relative errors. The 1% error threshold is highlighted for reference.

the network architecture, bit-width distribution, and FPGA implementation strategy.

The observed bit-width patterns also suggest possible directions for more targeted architectures. For example, deeper networks could be designed with strong precision constraints on the internal layers, while allowing only the final layer to retain higher bit-widths. Such a strategy would explicitly exploit the apparent robustness of the internal representation while preserving accuracy in the final reconstruction stage. Moreover, precision constraints could be chosen with the target hardware in mind, for instance by favoring LUT-based arithmetic below certain bit-widths and reserving DSP blocks for operations where their use is more efficient or necessary to meet timing.

A further point concerns the interpretation of the reported inference latency. The three-clock-cycle latency of the `hls4ml` core refers to a single evaluation of the NN at one radial position. However, in the intended control application the same equilibrium parameters must be evaluated over a fixed radial grid. For this reason, a more realistic FPGA design was also tested by encapsulating the NN core in a Vitis HLS wrapper that receives only the slowly varying parameters $(\alpha, \Theta_0, \Delta_H)$ through an Advanced eXtensible Interface 4 Lite (AXI4-Lite) interface, while the radial coordinate is generated internally from a fixed lookup table. The wrapper scans 21 predefined radial positions and streams the six reconstructed quantities for each point through an Advanced eXtensible Interface 4 Stream (AXI4-Stream) interface connected to direct memory access (DMA).

This design avoids transferring all radial input samples from the processor and minimizes communication overhead between the processing system and programmable logic. Because both the NN core and the radial scan loop are pipelined with II equal to one, one radial point can be processed at each clock cycle after the pipeline is filled. Post-implementation timing simulation confirmed a first-output latency of approx-

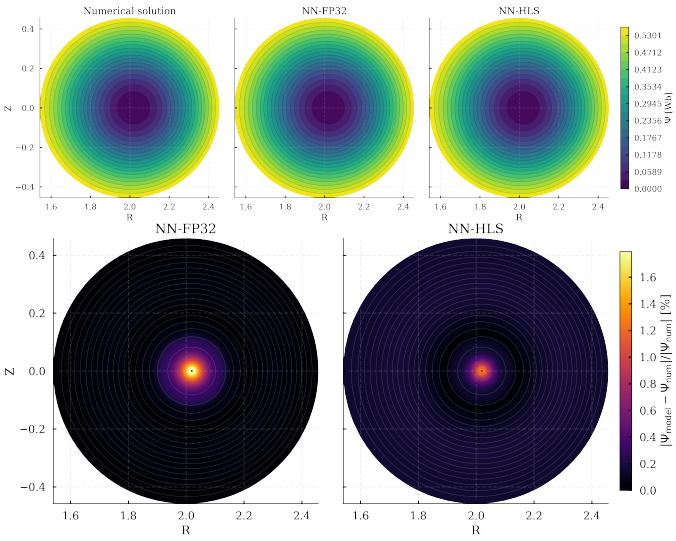

Fig. 7. Comparison of the reconstructed equilibrium poloidal flux $\Psi(R, Z)$ in the poloidal cross-section for the numerical solver, the FP32 model, and the selected quantized model. The bottom panels show the relative error of the neural surrogates with respect to the numerical solution.

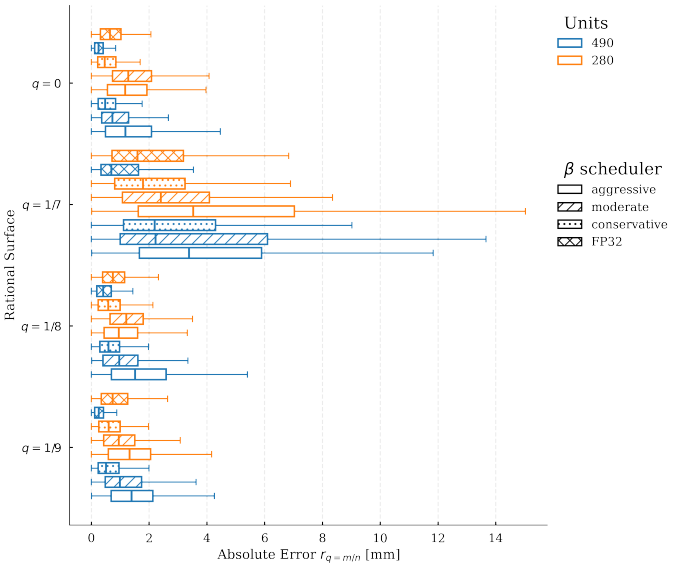

Fig. 8. Distribution of the rational-surface radial-position error over a subset of 500 RFP equilibria. The surfaces include the reversal surface $q = 0$ and the $m = 1$ rational surfaces $q = 1/7$, $q = 1/8$, and $q = 1/9$. The comparison highlights the effect of model size and quantization schedule on the localization accuracy of resonant surfaces.

imately 100 ns and a full-stream latency of approximately 300 ns for the complete 21-point radial scan at 100 MHz. The implemented design met timing for the target Kria K26 device with positive setup and hold slack, confirming that the accelerator can be integrated as a timing-closed low-latency programmable-logic module.

As a preliminary software reference, the same 21-point radial scan was also implemented on a central processing unit (CPU) using an allocation-free Numba [29] implementation in FP32 precision. The weights and biases were stored as contiguous arrays, the forward pass was manually implemented and compiled in `nopython` mode with `@njit(fastmath=True)`, and all intermediate buffers were preallocated and reused across calls. The first call, which triggers just-in-time compilation, was excluded from the measurement.

The optimized CPU implementation achieved an average latency of 86.2 μs for the full 21-point scan, with a standard deviation of 5.0 μs and a 99th percentile latency of 105.6 μs. This is close to the 100 μs period corresponding to the 10 kHz RFX-mod2 control cycle, leaving limited margin for additional computations, communication, and synchronization. In contrast, the post-implementation FPGA timing simulation indicates that the accelerator completes the same 21-point scan in approximately 300 ns, providing a substantially larger timing margin and deterministic execution.

Nevertheless, this comparison should be regarded as preliminary. The final deployment architecture will have to be assessed after extending the surrogate approach to the subsequent perturbation-reconstruction stage, since the complete real-time chain will include multiple NN blocks as well as additional data movement, synchronization, and control-system integration requirements.

Despite these application-specific limitations, the methodology developed in this work is not restricted to RFX-mod2. Although the numerical parameter ranges and equilibrium model used here are specific to RFX-mod2 RFP operation, the proposed workflow is more general. It consists of defining a bounded parametric physics solver, embedding known initial or boundary conditions in the surrogate architecture, performing multi-objective architecture selection, applying heterogeneous quantization-aware training, and evaluating candidate implementations under explicit latency and resource constraints. The same procedure can be transferred to other real-time reconstruction tasks in fusion devices whenever the relevant operating domain is known and deterministic low-latency inference is required.

VI. Conclusions

This work presented a compact quantized NN surrogate for real-time toroidal equilibrium reconstruction in RFX-mod2. Starting from the reference toroidal equilibrium model, a large synthetic database was generated over the range of parameters expected in RFP operation, allowing the network to learn the corresponding parametric solution map. The proposed formulation enforces the initial conditions by construction and accurately reconstructs the radial profiles of the equilibrium magnetic-field components and Shafranov shift.

A multi-objective architecture search was first used to identify compact floating-point models providing a favorable trade-off between accuracy and model size. The selected architectures were then optimized through quantization-aware training with heterogeneous precision assignment, using EBOPs as a proxy for hardware cost. The resulting quantized models showed that substantial reductions in arithmetic complexity can be achieved with limited degradation in reconstruction accuracy.

The optimized models were translated to FPGA firmware using `hls4ml` and synthesized and implemented for the

Kria K26 platform. All selected configurations achieved II equal to one clock cycle and inference latency of two to three clock cycles for a single radial evaluation. A wrapper-level implementation evaluating a 21-point radial grid reached a full-profile latency of approximately $300\,\mathrm{ns}$, substantially below the $100\,\mu\mathrm{s}$ period of the $10\,\mathrm{kHz}$ RFX-mod2 control cycle. The selected 280-conservative model provided the best compromise between accuracy and resource usage, while preserving physically relevant derived quantities such as the poloidal flux map and rational-surface positions.

These results demonstrate that a compact, quantized neural surrogate can reproduce the toroidal equilibrium quantities required for subsequent perturbation reconstruction while satisfying stringent real-time constraints. Future work will extend the same hardware-aware modelling approach to the full toroidal magnetic-perturbation reconstruction chain and assess its integration into the RFX-mod2 real-time control system.

ACKNOWLEDGMENT

The authors acknowledge the use of ChatGPT based on GPT-5.2, developed by OpenAI, for grammar checking, language refinement, and suggestions aimed at improving the clarity and consistency of the manuscript. The authors reviewed and edited all AI-assisted suggestions and take full responsibility for the scientific content, analysis, conclusions, and final form of the manuscript [30].

REFERENCES

[1] L. Marrelli, R. Cavazzana, D. Bonfiglio, M. Gobbin, G. Marchiori, S. Peruzzo *et al.*, "Upgrades of the RFX-mod reversed field pinch and expected scenario improvements," *Nuclear Fusion*, vol. 59, no. 7, p. 076027, Jun. 2019. [Online]. Available: https://doi.org/10.1088/1741-4326/ab1c6a

[2] D. Terranova, M. Agostini, F. Auriemma, M. Gobbin, G. Marchiori, L. Pigatto *et al.*, "RFX-mod2 as a flexible device for reversed-field-pinch and low-field tokamak research," *Nuclear Fusion*, vol. 64, no. 7, p. 076003, May 2024. [Online]. Available: https://doi.org/10.1088/1741-4326/ad4805

[3] L. Carraro, M. Zuin, D. Abate, P. Agostinetti, M. Agostini, D. Aprile *et al.*, "RFX-mod2 diagnostic capability enhancements for the exploration of multi-magnetic-configurations," *Nuclear Fusion*, vol. 64, no. 7, p. 076032, Jun. 2024. [Online]. Available: https://doi.org/10.1088/1741-4326/ad490a

[4] N. Marconato, P. Bettini, R. Cavazzana, L. Grando, G. Marchiori, L. Marrelli *et al.*, "Design of the new electromagnetic measurement system for RFX-mod upgrade," *Fusion Engineering and Design*, vol. 146, pp. 906–909, Sep. 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S092037961930122X

[5] G. Manduchi, A. Luchetta, C. Taliercio, A. Rigoni, G. Martini, R. Cavazzana *et al.*, "The upgrade of the control and data acquisition system of RFXMod2," *Fusion Engineering and Design*, vol. 167, p. 112329, Jun. 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0920379621001058

[6] L. Marrelli, P. Martin, M. Puiatti, J. Sarff, B. Chapman, J. Drake *et al.*, "The reversed field pinch," *Nuclear Fusion*, vol. 61, no. 2, p. 023001, Jan. 2021. [Online]. Available: https://doi.org/10.1088/1741-4326/abc06c

[7] P. Zanca and D. Terranova, "Reconstruction of the magnetic perturbation in a toroidal reversed field pinch," *Plasma Physics and Controlled Fusion*, vol. 46, no. 7, p. 1115, Jun. 2004. [Online]. Available: https://dx.doi.org/10.1088/0741-3335/46/7/011

[8] J. Lister and H. Schnurrenberger, "Fast non-linear extraction of plasma equilibrium parameters using a neural network mapping," *Nuclear Fusion*, vol. 31, no. 7, p. 1291, Jul. 1991. [Online]. Available: https://doi.org/10.1088/0029-5515/31/7/005

[9] J. Kates-Harbeck, A. Svyatkovskiy, and W. Tang, "Predicting disruptive instabilities in controlled fusion plasmas through deep learning," *Nature*, vol. 568, no. 7753, pp. 526–531, Apr. 2019. [Online]. Available: https://www.nature.com/articles/s41586-019-1116-4

[10] L. Orlandi, A. Rigoni Garola, M. Gobbin, P. Franz, and L. Piron, "Data reconstruction using variational autoencoders and error analysis compared to B-spline interpolation," *Plasma Physics and Controlled Fusion*, vol. 68, no. 2, p. 025028, Feb. 2026. [Online]. Available: https://doi.org/10.1088/1361-6587/ae4716

[11] K. L. van de Plassche, J. Citrin, C. Bourdelle, Y. Camenen, F. J. Casson, V. I. Dagnelie *et al.*, "Fast modeling of turbulent transport in fusion plasmas using neural networks," *Physics of Plasmas*, vol. 27, no. 2, p. 022310, Feb. 2020. [Online]. Available: https://doi.org/10.1063/1.5134126

[12] J. Degrave, F. Felici, J. Buchli, M. Neunert, B. Tracey, F. Carpanese *et al.*, "Magnetic control of tokamak plasmas through deep reinforcement learning," *Nature*, vol. 602, no. 7897, pp. 414–419, Feb. 2022. [Online]. Available: https://www.nature.com/articles/s41586-021-04301-9

[13] J. Duarte, S. Han, P. Harris, S. Jindariani, E. Kreinar, B. Kreis *et al.*, "Fast inference of deep neural networks in FPGAs for particle physics," *Journal of Instrumentation*, vol. 13, no. 07, pp. P07 027–P07 027, Jul. 2018. [Online]. Available: http://arxiv.org/abs/1804.06913

[14] T. Aarrestad, V. Loncar, N. Ghielmetti, M. Pierini, S. Summers, J. Ngadiuba *et al.*, "Fast convolutional neural networks on FPGAs with hls4ml," *Machine Learning: Science and Technology*, vol. 2, no. 4, p. 045015, Dec. 2021. [Online]. Available: http://arxiv.org/abs/2101.05108

[15] E. E. Khoda, D. Rankin, R. Teixeira de Lima, P. Harris, S. Hauck, S.-C. Hsu *et al.*, "Ultra-low latency recurrent neural network inference on FPGAs for physics applications with hls4ml," *Machine Learning: Science and Technology*, vol. 4, no. 2, p. 025004, Apr. 2023. [Online]. Available: https://doi.org/10.1088/2632-2153/acc0d7

[16] C. N. Coelho, A. Kuusela, S. Li, H. Zhuang, T. Aarrestad, V. Loncar *et al.*, "Automatic heterogeneous quantization of deep neural networks for low-latency inference on the edge for particle detectors," *Nature Machine Intelligence*, vol. 3, no. 8, pp. 675–686, Jun. 2021. [Online]. Available: http://arxiv.org/abs/2006.10159

[17] C. Sun, T. K. Årrestad, V. Loncar, J. Ngadiuba, and M. Spiropulu, "Gradient-based Automatic Mixed Precision Quantization for Neural Networks On-Chip," Aug. 2024. [Online]. Available: http://arxiv.org/abs/2405.00645

[18] A. Rigoni Garola, R. Cavazzana, M. Gobbin, R. S. Delogu, G. Manduchi, C. Taliercio *et al.*, "Diagnostic Data Integration Using Deep Neural Networks for Real-Time Plasma Analysis," *IEEE Transactions on Nuclear Science*, vol. 68, no. 8, pp. 2165–2172, Aug. 2021. [Online]. Available: https://ieeexplore.ieee.org/document/9481901

[19] Y. Wei, R. F. Forelli, C. Hansen, J. P. Levesque, N. Tran, J. C. Agar *et al.*, "Low latency optical-based mode tracking with machine learning deployed on FPGAs on a tokamak," *Review of Scientific Instruments*, vol. 95, no. 7, p. 073509, Jul. 2024. [Online]. Available: https://doi.org/10.1063/5.0190354

[20] F. Chollet *et al.*, "Keras," 2024. [Online]. Available: https://github.com/keras-team/keras

[21] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean *et al.*, "TensorFlow: A system for large-scale machine learning," May 2016. [Online]. Available: http://arxiv.org/abs/1605.08695

[22] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, "Optuna: A Next-generation Hyperparameter Optimization Framework," in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, ser. KDD '19. New York, NY, USA: Association for Computing Machinery, Jul. 2019, pp. 2623–2631. [Online]. Available: https://dl.acm.org/doi/10.1145/3292500.3330701

[23] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, Apr. 2002. [Online]. Available: https://ieeexplore.ieee.org/document/996017

[24] FastML Team, "fastmachinelearning/hls4ml," 2024. [Online]. Available: https://github.com/fastmachinelearning/hls4ml

[25] AMD Xilinx, *Vivado Design Suite User Guide*, AMD, 2024, version 2024.1. [Online]. Available: https://docs.amd.com/r/2024.1-English/ug973-vivado-release-notes-install-license/Release-Notes

[26] ——, *Kria KR260 Robotics Starter Kit*, AMD, 2024, accessed 2026. [Online]. Available: https://www.amd.com/en/products/system-on-modules/kria/k26/kr260-robotics-starter-kit.html

[27] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A Simple Way to Prevent Neural Networks from Overfitting," *Journal of Machine Learning Research*,

vol. 15, no. 56, pp. 1929–1958, 2014. [Online]. Available: http://jmlr.org/papers/v15/srivastava14a.html

[28] M. Raissi, P. Perdikaris, and G. E. Karniadakis, "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations," *Journal of Computational Physics*, vol. 378, pp. 686–707, Feb. 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0021999118307125

[29] S. K. Lam, A. Pitrou, and S. Seibert, "Numba: A LLVM-based Python JIT compiler," in *Proceedings of the Second Workshop on the LLVM Compiler Infrastructure in HPC*, ser. LLVM '15. New York, NY, USA: Association for Computing Machinery, Nov. 2015, pp. 1–6. [Online]. Available: https://dl.acm.org/doi/10.1145/2833157.2833162

[30] OpenAI, "Introducing gpt-5.2," 2025. [Online]. Available: https://openai.com/index/introducing-gpt-5-2/