Supplementary Information

Scalable Microring-Based Interconnects with Channel Equalization and Closed-Loop Stabilization

Zichun Lin†, Zhican Zhou†, Xiangpeng Ou, Weiping Li, Xuhao Wu, Hongyan Yu, Xiaoyi Song, William He, and Yating Wan*

Integrated Photonics Lab, CEMSE, King Abdullah University of Science and Technology, Thuwal, Makkah Region 23955, Saudi Arabia

†These authors contributed equally to this work.

*Correspondence: yating.wan@kaust.edu.sa

Contents

1 Supplementary Note 1: Device parameters **2**

2 Supplementary Note 2: Routing capability and scalability of the MRR switching fabric **4**

3 Supplementary Note 3: FPGA-assisted control algorithm **6**

4 Supplementary Note 4: Hardware-aware simulation **8**

4.1 Static channel-loss model 9

4.2 Layer-wise hardware impairments 11

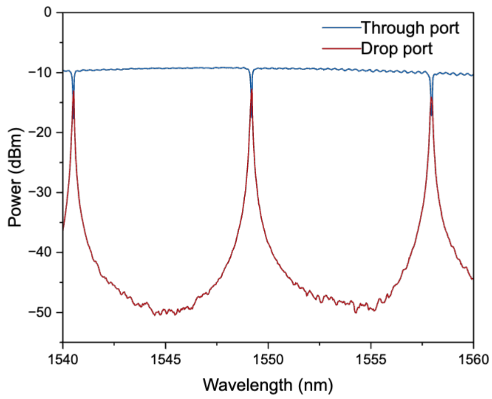
4.3 Layer-wise feature-map error 11

4.4 Simulation settings and representative results 12

5 Supplementary References **13**

1 Supplementary Note 1: Device parameters

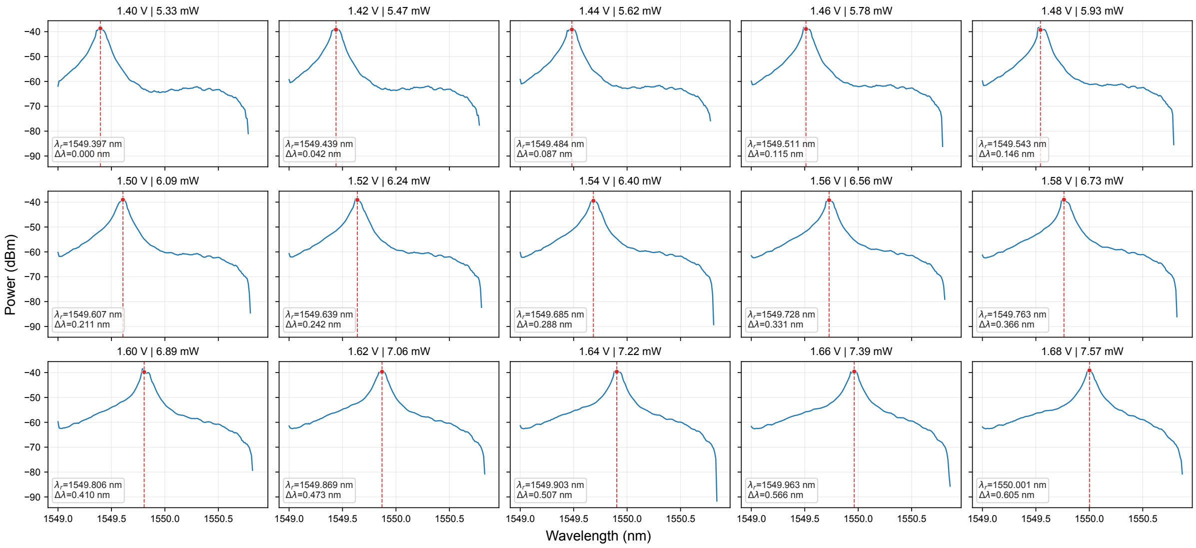
The passive spectral response of a representative microring resonator (MRR) is shown in Supplementary Fig. 1. The through-port spectrum exhibits periodic resonance notches, while the drop-port spectrum shows the corresponding resonance peaks. From the wavelength spacing between adjacent resonances, the measured free spectral range (FSR) is extracted to be 8.715 nm. The average quality factor extracted from the measured resonances is $Q = 17{,}387$, confirming the wavelength selectivity required for wavelength-division multiplexing (WDM) routing and channel-power equalization.



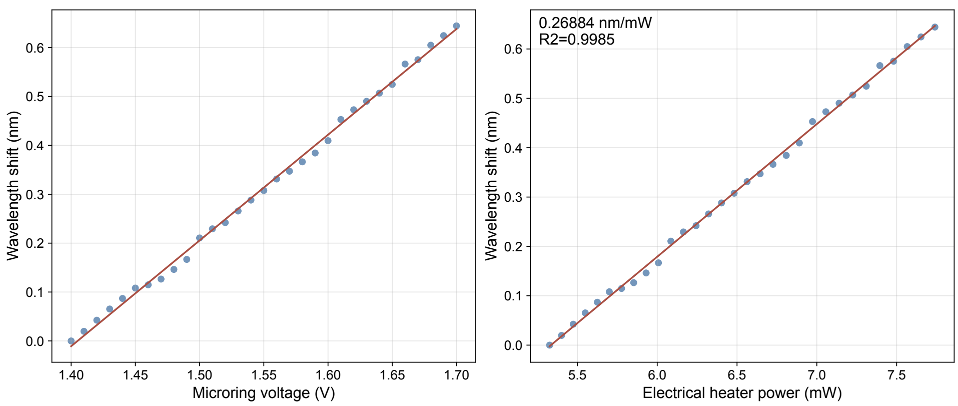
Supplementary Fig. 1: Measured through-port and drop-port spectra of a representative MRR. The extracted free spectral range is 8.715 nm, and the average resonance quality factor is $Q = 17{,}387$.

The thermo-optic tuning characteristics of the MRRs were extracted from the measured transmission spectra under different heater-driving conditions. Supplementary Fig. 2 shows the measured resonance spectra as the heater voltage is varied. As the electrical heater power increases, the MRR resonance progressively shifts toward longer wavelengths due to the thermo-optic effect [1]. For each heater condition, the resonance wavelength is extracted from the measured spectral response, and the wavelength shift is calculated relative to the initial resonance position. These extracted resonance positions are then used to quantify the heater tuning efficiency.

Supplementary Fig. 3 summarizes the extracted resonance wavelength shifts as a function of the applied microring voltage and the corresponding electrical heater power. The



Supplementary Fig. 2: Measured resonance spectra of the MRR under different heater-driving conditions. The resonance position shifts with the applied heater voltage due to the thermo-optic effect.

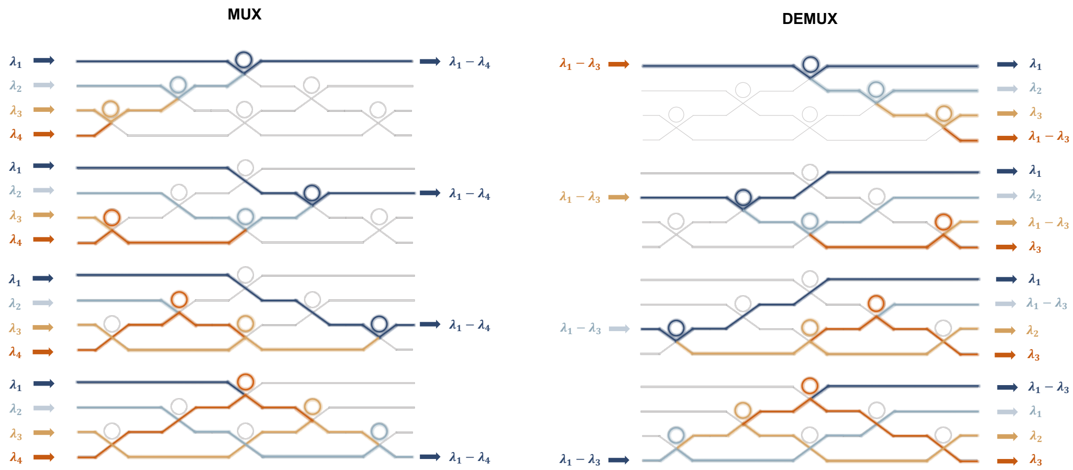


Supplementary Fig. 3: Thermo-optic tuning characteristics of the MRR heater. Left: extracted resonance wavelength shift as a function of the applied microring voltage. Right: extracted resonance wavelength shift as a function of the electrical heater power. The linear fit gives a thermo-optic tuning efficiency of 0.26884 nm/mW with $R^2 = 0.9985$.

resonance shift increases monotonically with heater voltage and approximately linearly with electrical heater power within the measured operating range. A linear fit of the wavelength shift as a function of heater power gives a thermo-optic tuning efficiency of 0.26884 nm/mW with an R^2 value of 0.9985, confirming the highly linear thermal tuning response. Using the measured heater resistance of approximately 300 Ω, the electrical power of each thermal phase shifter is estimated as $P_i = V_i^2/R_i$. For the representative routing configurations, summing the heater powers of the active routing MRRs gives total routing control powers of 70.6 mW for the wavelength-multiplexing (MUX) state and 58.2 mW for the wavelength-demultiplexing (DEMUX) state. These values represent the electrical heater power consumed by the routing stage and do not include the external field-programmable gate array (FPGA), analog-to-digital converter / digital-to-analog converter (ADC/DAC) electronics, or the thermoelectric cooler (TEC) used for temperature-perturbation measurements.

2 Supplementary Note 2: Routing capability and scalability of the MRR switching fabric

Supplementary Fig. 4 illustrates representative wavelength-selective routing configurations supported by the proposed MRR-based switching fabric. The left column shows wavelength multiplexing (MUX) configurations, in which four wavelength channels launched from multiple input ports are selectively routed and combined toward a common output port. The right column shows wavelength demultiplexing (DEMUX) configurations, in which multiwavelength signals launched from a common input port are wavelength-selectively separated and directed to different output ports according to the programmed MRR resonance states. In the DEMUX configurations, three active wavelength channels are used, while the remaining port is reserved for readout or monitoring. This setting is adopted because practical MRR-based routing exhibits finite extinction ratio, insertion loss, and residual through-port leakage, rather than an ideal response with perfect suppression of the selected wavelength in the through path. Using three active DEMUX channels therefore reduces ambiguity from residual leakage and provides a clearer verification of wavelength-selective separation. In each configuration, colored paths indicate the selected wavelength channels, whereas gray paths indicate inactive or unselected routing paths.



Supplementary Fig. 4: Representative MUX and DEMUX routing configurations of the MRR-based switching fabric. Colored paths denote the actively routed wavelength channels, and gray paths denote inactive or unselected routing paths.

The scalability of the proposed MRR-based switching fabric can be evaluated from the number of selectable routing assignments supported by different routing functions. Here, N denotes the number of input/output ports, K denotes the number of selected wavelength channels, and $P(N, K) = N!/(N-K)!$ represents the number of ordered assignments of K distinct ports selected from N ports. Supplementary Table 1 summarizes the representative routing functions supported by the switching fabric, including single-wavelength routing, wavelength multiplexing, and wavelength demultiplexing. These routing functions provide the basis for flexible inter-PCU optical signal transfer in distributed photonic computing systems.

For the demonstrated 4×4 implementation, the fabric supports $16K$ selectable single-wavelength routes, and 96 MUX or DEMUX assignments when four wavelength channels are considered. These counts represent the number of conflict-free routing assignments under the stated assumptions.

Supplementary Table 1: **Representative Routing Functions Supported by the MRR-Based Switching Fabric**

Routing function	Operation	Supported count	Description
Routing	$I_i(\lambda_k) \to O_j$	N^2K	A selected wavelength channel can be routed from any input port to any output port. For the demonstrated 4×4 fabric, this corresponds to $16K$ selectable single-channel routes.
MUX	$\{I_i(\lambda_k)\} \to O_j$	$NP(N,K)$	K wavelength channels from selected input ports are combined toward one output port. For $N = 4$ and $K = 4$, this gives $4 \times 4! = 96$ assignments.
DEMUX	$I_i(\{\lambda_k\}) \to \{O_j(\lambda_k)\}$	$NP(N,K)$	K wavelength channels from one input port are separated and directed to selected output ports. For $N = 4$ and $K = 4$, this gives $4 \times 4! = 96$ assignments.

3 Supplementary Note 3: FPGA-assisted control algorithm

The FPGA-assisted PID locking scheme used in this work, referred to as TP-FAPL, consists of three stages, as summarized in Algorithm 1. The objective is to stabilize the transmitted optical power of a selected MRR channel at a target value P_{tar} by updating the heater-driving DAC code in real time.

In Supplementary Algorithm 1, λ_i denotes the selected wavelength channel, P_{tar} is the target optical power, and $\mathcal{L}$ represents the pre-characterized voltage–power lookup table. For each selected wavelength channel, the corresponding LUT $\mathcal{L}_i(u)$ provides the accessible power range and the voltage–power mapping used to guide target acquisition. The variable u_k is the DAC code applied to the MRR heater at the k-th control update, while P_k is the corresponding optical power measured by the ADC. The function $\mathrm{sat}_{[a,b]}(\cdot)$ limits the control variable to the range $[a, b]$.

In Phase 1, the LUT is constructed or loaded for the selected wavelength channel, but no final operating point is directly assigned from the LUT. In Phase 2, the controller uses the LUT-defined target range and real-time ADC feedback to perform monotonic target acquisition. Rather than directly applying a LUT-inferred DAC code, the DAC code is increased from the low-DAC side until the measured power crosses the target level. The crossing condition depends on the optical port configuration: in drop-port operation, the search stops when the ADC value rises above the target; in through-port

Supplementary Algorithm 1 Three-phase FPGA-assisted PID locking for MRR channel-power stabilization

procedure TP-FAPL($\lambda_i, P_{\text{tar}}, \mathcal{L}, K_p, K_i, K_d$)
    Set DAC bounds $[u_{\min}, u_{\max}]$, search step Δu, backoff B, and integral bound $I_{\max}$
    Initialize $k \leftarrow 0$, $I_0 \leftarrow 0$, $e_{-1} \leftarrow 0$

    Phase 1: voltage–power LUT construction
    Construct or load the voltage–power LUT $\mathcal{L}_i(u)$ for wavelength channel λ_i
    Obtain the accessible power range $[P_{\min}, P_{\max}]$ from $\mathcal{L}_i(u)$
    Verify $P_{\text{tar}} \in [P_{\min}, P_{\max}]$ and set $u_0 \leftarrow u_{\min}$

    Phase 2: LUT-assisted monotonic target acquisition
    while target crossing is not reached and $u_k < u_{\max}$ **do**
        Apply u_k to the heater DAC and read P_k from the ADC
        if drop-port mode and $P_k \geq P_{\text{tar}}$ **then**
            break
        else if through-port mode and $P_k \leq P_{\text{tar}}$ **then**
            break
        end if
        $u_{k+1} \leftarrow \text{sat}_{[u_{\min}, u_{\max}]}\,(u_k + \Delta u), \quad k \leftarrow k + 1$
    end while
    $u_k \leftarrow \text{sat}_{[u_{\min}, u_{\max}]}\,(u_k - B\Delta u)$
    Apply the backed-off DAC code and read the updated power P_k
    Initialize branch polarity σ_k according to the optical port configuration

    Phase 3: sign-aware PID locking with anti-windup
    while feedback locking is enabled **do**
        Read P_k and update σ_k from recent DAC and ADC changes when available
        $e_k \leftarrow \sigma_k\,(P_{\text{tar}} - P_k)$
        $I_k^{\text{cand}} \leftarrow I_{k-1} + K_i e_k$
        if integral update is allowed **then**
            $I_k \leftarrow \text{sat}_{[-I_{\max}, I_{\max}]}\left(I_k^{\text{cand}}\right)$
        else
            $I_k \leftarrow I_{k-1}$
        end if
        $u_{k+1} \leftarrow \text{sat}_{[u_{\min}, u_{\max}]}\,[u_k + K_p e_k + I_k + K_d(e_k - e_{k-1})]$
        Apply u_{k+1} to the heater DAC; $\quad e_{k-1} \leftarrow e_k$, $k \leftarrow k + 1$
    end while
end procedure

operation, it stops when the ADC value falls below the target. This procedure avoids direct dependence on a pre-characterized bias point and provides a robust initialization for closed-loop locking.

In Phase 3, the local branch polarity σ_k is introduced for sign-aware PID locking. The required DAC correction direction depends on the selected resonance branch and optical port, so the raw power error is multiplied by σ_k before the PID update. The controller also includes anti-windup protection by conditionally updating the integral term, clamping the integral accumulator, and saturating the DAC output within its hardware range. These operations preserve the correct feedback direction on different resonance branches and prevent excessive integral accumulation when the control output approaches its hardware limits.

4 Supplementary Note 4: Hardware-aware simulation

We performed hardware-aware inference simulations on CIFAR-10 [2] using a clean-training / hardware-aware-testing protocol to evaluate how static channel-power mismatch propagates through convolutional neural networks and how channel equalization suppresses the resulting feature-map distortion. Two representative architectures were considered: VGG-11 [3] and ResNet-18 [4]. Both networks were trained under clean 32-bit floating-point (FP32) digital conditions and were not retrained under optical channel mismatch. During inference, each convolutional module was augmented with a hardware-aware channel-loss module, which applied channel-dependent optical-power attenuation to the input feature channels before convolution. Input noise and simulated low-bit quantization were also injected only at test time.

All CIFAR-10 images were processed using the same preprocessing pipeline: resizing to 256×256, center cropping to 224×224, conversion to tensors, and normalization with mean $(0.4914, 0.4822, 0.4465)$ and standard deviation $(0.2023, 0.1994, 0.2010)$. For the layer-wise feature-map error analysis, $N = 256$ CIFAR-10 test images were used.

VGG-11 was analyzed over its eight convolutional layers. ResNet-18 was analyzed over twenty convolutional modules, including the initial convolution, all residual-block convolutions, and the projection/downsample convolutions. Feature maps were recorded at the outputs of the wrapped convolutional modules, i.e., after the convolution operation and before the subsequent batch normalization, nonlinear activation, or residual addition when applicable.

4.1 Static channel-loss model

Let x_ℓ be the input activation tensor to convolutional layer ℓ. To emulate channel-power nonuniformity in the optical interconnect, each input channel is multiplied by a static optical-power gain before the corresponding convolution. The experimentally measured unequalized optical-power readings were first converted into relative channel-loss classes by referencing the strongest channel. For the measured transmission levels -13.46, -15.17, and -19.96 dB, this gives the measurement-derived relative loss classes

$$\mathbf{d}^{(0)} = [0,\ 1.71,\ 6.50]\ \text{dB}. \tag{1}$$

The corresponding measured mismatch span is defined as

$$L_0 = \max_i d_i^{(0)} - \min_i d_i^{(0)} = 6.5\ \text{dB}. \tag{2}$$

For a requested maximum mismatch bound L, the relative loss classes were scaled as

$$d_i(L) = d_i^{(0)} \frac{L}{L_0}. \tag{3}$$

For each convolutional layer, input-channel losses were sampled from these measurement-derived classes with equal probability. A small dB-domain Gaussian jitter was added to emulate device-to-device variation:

$$\ell_{\ell,c}^{(r)} = \text{clip}\left(d_{z_{\ell,c}}(L) + \epsilon_{\ell,c}^{(r)},\ 0,\ L\right), \qquad \epsilon_{\ell,c}^{(r)} \sim \mathcal{N}\left(0, \sigma_j^2(L)\right), \tag{4}$$

where $z_{\ell,c}$ is the sampled loss-class index, c is the input-channel index, and r is the Monte-Carlo channel-loss realization. The jitter standard deviation is scaled with the requested mismatch bound as

$$\sigma_j(L) = \sigma_{j,0}\frac{L}{L_0}, \tag{5}$$

where $\sigma_{j,0} = 0.3$ dB is the jitter standard deviation at the measured mismatch span L_0. The sampled loss is clipped to the requested mismatch interval $[0, L]$.

The corresponding optical-power gain is

$$g_{\ell,c}^{(r)} = 10^{-\ell_{\ell,c}^{(r)}/10}. \tag{6}$$

To focus on channel-to-channel nonuniformity rather than accumulated common insertion loss, a scalar layer-wise global-power compensation was applied by normalizing the sampled power gains by their arithmetic mean:

$$\hat{g}_{\ell,c}^{(r)} = \frac{g_{\ell,c}^{(r)}}{\frac{1}{C_\ell}\sum_{c'=1}^{C_\ell} g_{\ell,c'}^{(r)}}, \tag{7}$$

where C_ℓ is the number of input channels of layer ℓ. The compensated gain was applied uniformly to all spatial positions and all test images within that Monte-Carlo realization:

$$\tilde{x}_{\ell,n,c,i,j}^{(r)} = x_{\ell,n,c,i,j}\hat{g}_{\ell,c}^{(r)}. \tag{8}$$

This model represents a static wavelength- or route-dependent transmission mismatch in a routed WDM photonic interconnect, rather than image-dependent random noise.

For the equalized condition, the residual channel-power loss was modeled as a small uniform residual mismatch bounded by the experimentally achieved post-equalization variation:

$$\ell_{\ell,c,\mathrm{EQ}}^{(r)} \sim \mathrm{Uniform}(0, 0.2\text{ dB}). \tag{9}$$

Thus, EQ was represented as suppression of the residual test-time channel-power mismatch, rather than as retraining or modification of the neural-network weights.

4.2 Layer-wise hardware impairments

In addition to the static channel-power mismatch, all clean, unequalized, and equalized inference conditions included the same calibrated hardware front-end impairments. Additive Gaussian input noise was applied at each wrapped convolutional input, with standard deviation equal to 5% of the clean no-mismatch calibrated layer-input root-mean-square (RMS). The noise standard deviation was calibrated once using clean no-mismatch inference samples and then shared across clean, unequalized, and equalized evaluation.

Input activations were quantized using 5-bit symmetric quantization. For each convolutional input tensor, the quantization scale was calibrated from clean no-mismatch activations using 512 calibration samples. The dynamic range was set by the 99.9th percentile of the absolute activation values, so that rare outliers did not dominate the quantization scale. Convolution weights were quantized using 5-bit symmetric quantization with per-output-channel dynamic scales, while biases remained in FP32. For layer-wise L_2 comparisons, the clean and perturbed feature-map passes used paired random seeds for the hardware noise, so that the measured feature-map difference primarily reflects the channel-mismatch condition rather than independent noise draws.

4.3 Layer-wise feature-map error

For each convolutional layer ℓ, mismatch condition $m \in \{\mathrm{noEQ}, \mathrm{EQ}\}$, and Monte-Carlo realization r, we recorded the clean hardware-reference feature map $Y_\ell^{(0)}$ and the perturbed feature map $\tilde{Y}_\ell^{(m,r)}$. The layer-wise relative L_2-norm error was computed over the $N = 256$ test samples as

$$e_\ell^{(r)}(m) = \frac{\left(\sum_{n=1}^{N} \left\|\tilde{Y}_{\ell,n}^{(m,r)} - Y_{\ell,n}^{(0)}\right\|_2^2\right)^{1/2}}{\left(\sum_{n=1}^{N} \left\|Y_{\ell,n}^{(0)}\right\|_2^2\right)^{1/2} + 10^{-12}}. \tag{10}$$

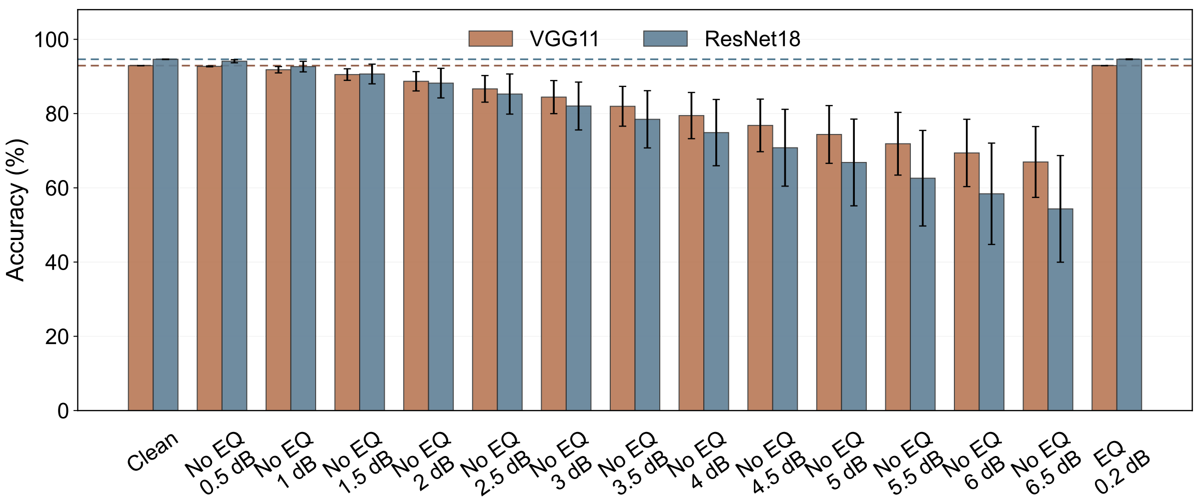
The reported layer-wise error was averaged over $R = 5$ Monte-Carlo channel-loss realizations:

$$e_\ell(m) = \frac{1}{R} \sum_{r=1}^{R} e_\ell^{(r)}(m). \tag{11}$$

Accuracy was evaluated on the full CIFAR-10 test set using the same Monte-Carlo seed set.

4.4 Simulation settings and representative results

For the unequalized mismatch sweep, both VGG-11 and ResNet-18 were evaluated using measurement-inspired channel-loss distributions scaled to maximum mismatch bounds of $L = 0, 0.5, 1.0, \ldots, 6.5$ dB. The representative unequalized condition reported in the main text corresponds to the experimentally measured maximum mismatch span, $L = 6.5$ dB. The equalized condition used a residual mismatch bounded by 0.2 dB, consistent with the experimentally achieved post-equalization channel-power variation.



Supplementary Fig. 5: Hardware-aware CIFAR-10 inference accuracy of VGG-11 and ResNet-18 under channel-power mismatch. Error bars denote the standard deviation over $R = 5$ repeated hardware-mismatch realizations.

Supplementary Fig. 5 summarizes the full-test-set CIFAR-10 inference accuracy of VGG-11 and ResNet-18 under clean, unequalized, and equalized channel-power conditions. In the clean hardware-reference condition, the models achieve accuracies of 92.92% and 94.63%, respectively. As the maximum unequalized channel-power mismatch increases from 0.5 to 6.5 dB, both models exhibit a progressive accuracy degradation, confirming that channel-dependent optical-power errors can propagate through the convolutional layers and impair network inference. The degradation is more pronounced for ResNet-18 at large mismatch levels, where the accuracy decreases to 54.33% at $L = 6.5$ dB. In contrast, the equalized case with a residual mismatch bounded by 0.2 dB recovers the accuracy to near the clean hardware-reference level for both networks.

5 Supplementary References

References

[1] W. Bogaerts, P. De Heyn, T. Van Vaerenbergh, K. De Vos, S. Kumar Selvaraja, T. Claes, P. Dumon, P. Bienstman, D. Van Thourhout, and R. Baets. Silicon microring resonators. *Laser & Photonics Reviews*, 6(1):47–73, 2012.

[2] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

[3] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[4] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.