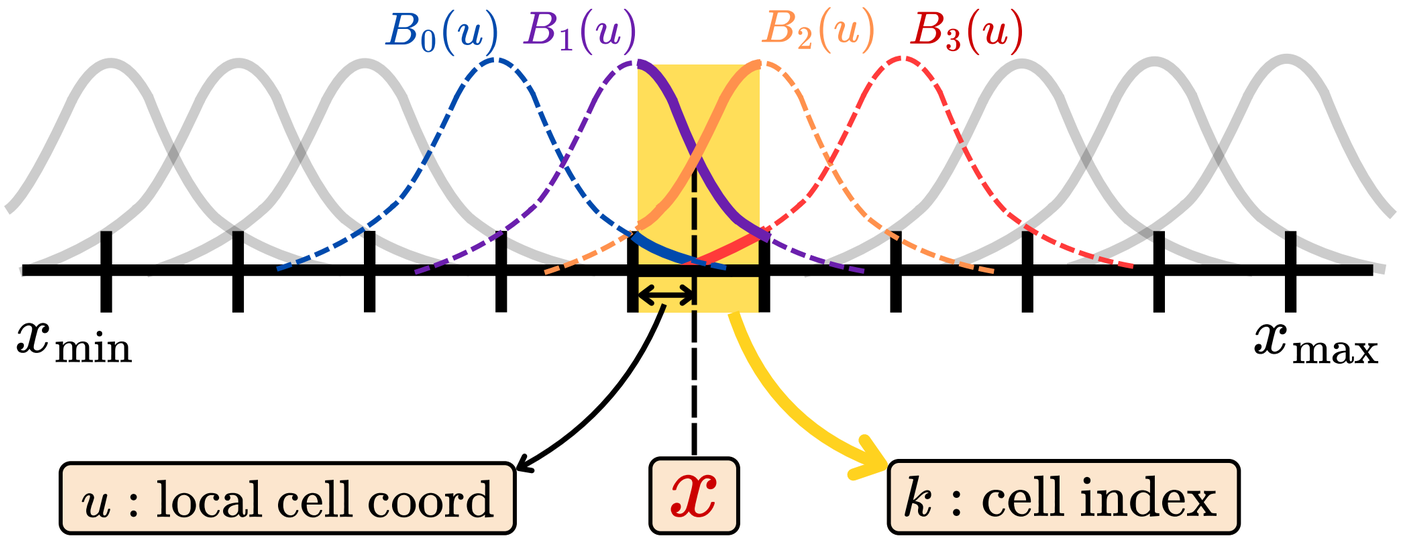
Ultrafast On-Chip Online Learning via Spline Locality in Kolmogorov-Arnold Networks

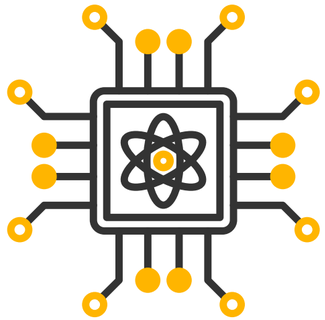


Duc Hoang*, **Aarush Gupta***, Philip Harris

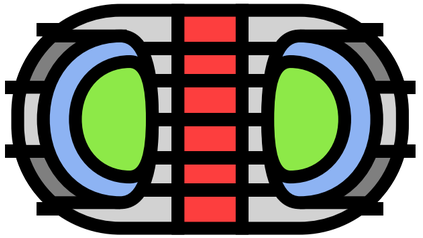
MIT

*equal contribution

Phil Harris Lab@MIT

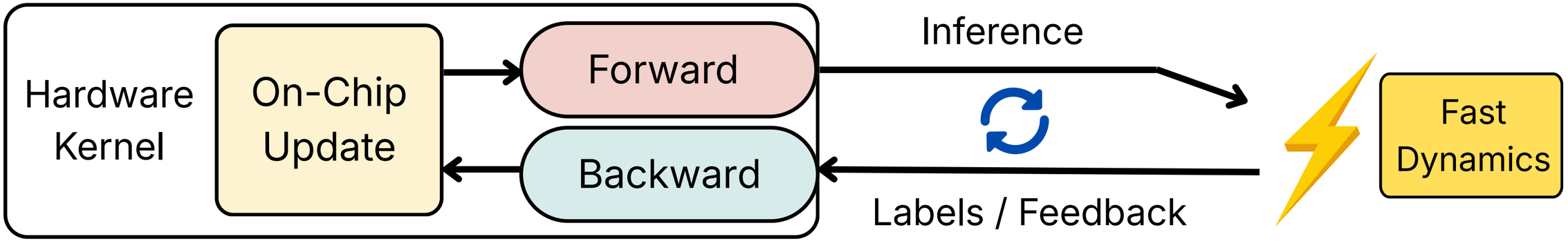


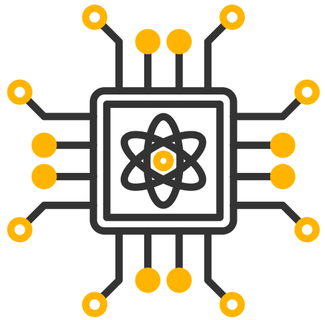
Quantum Computing



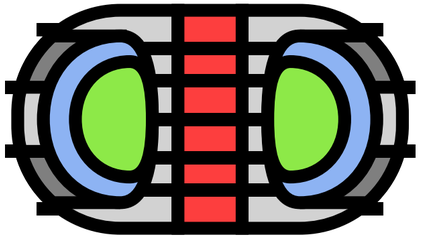
Fusion Control

Communication



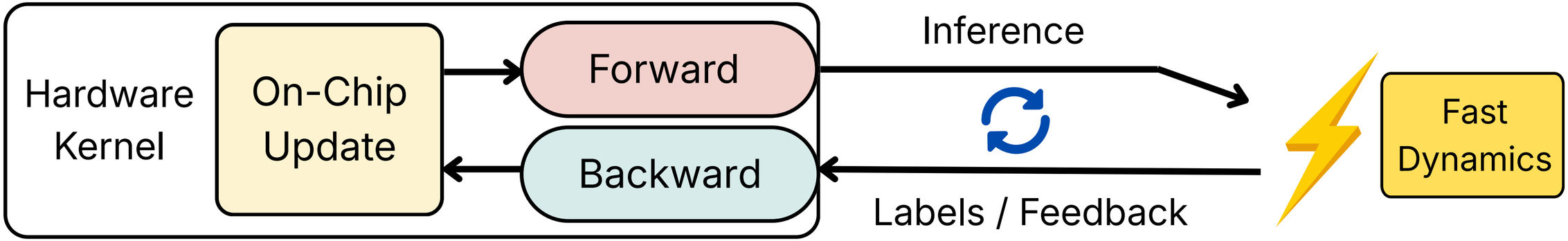


Quantum Computing

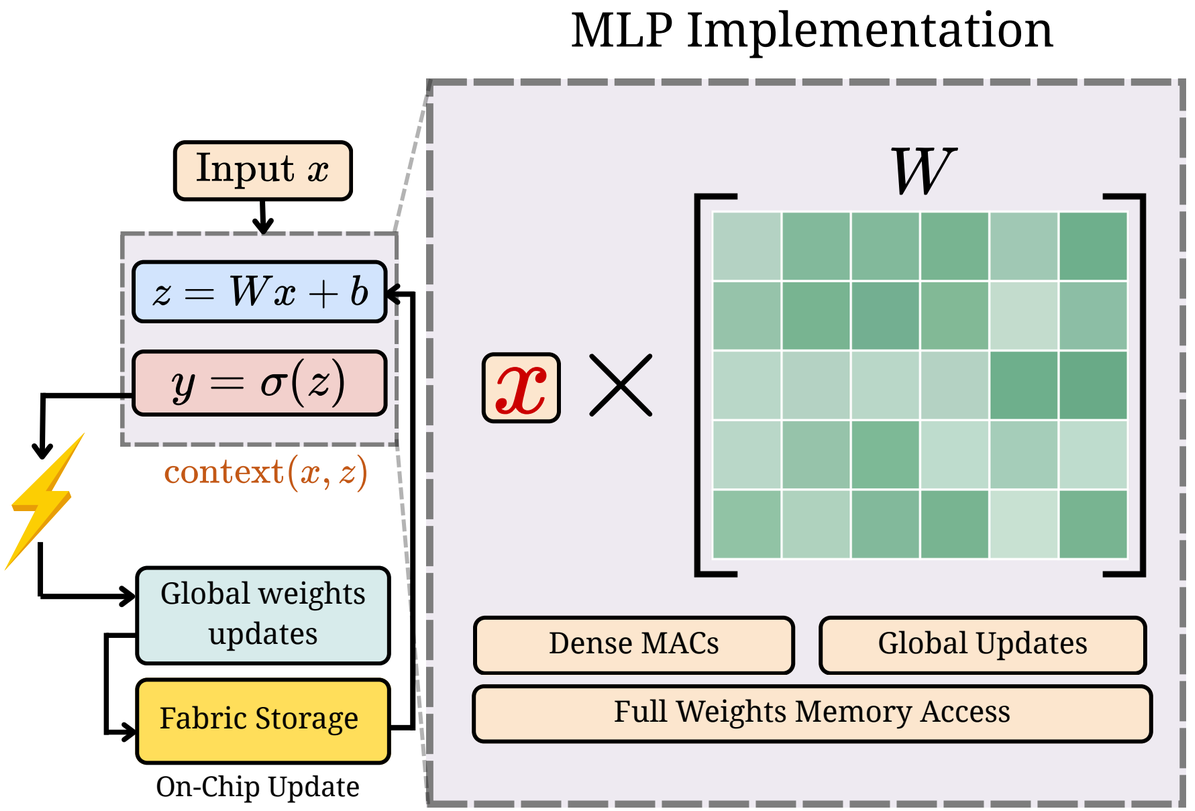


Fusion Control

Communication



Sub-microsecond update latency needed! (FPGA / ASIC)


MLP Implementation
Input x
$z = Wx + b$
$y = \sigma(z)$
$\text{context}(x, z)$
Global weights updates
Fabric Storage
On-Chip Update
x
W
Dense MACs
Global Updates
Full Weights Memory Access

MLP $y_q = \phi\left(\sum_p w_{q,p} x_p\right)$

MLP $y_q = \phi\left(\sum_p w_{q,p} x_p\right)$

learned activations replace learned weights

KAN $y_q = \sum_p \phi_{q,p}(x_p)$

MLP $y_q = \phi\left(\sum_p w_{q,p} x_p\right)$

learned activations replace learned weights

KAN $y_q = \sum_p \phi_{q,p}(x_p)$ where $\phi_{q,p}(x) = \sum_i c_{q,p,i} B_i(x)$

we learn these coefficients

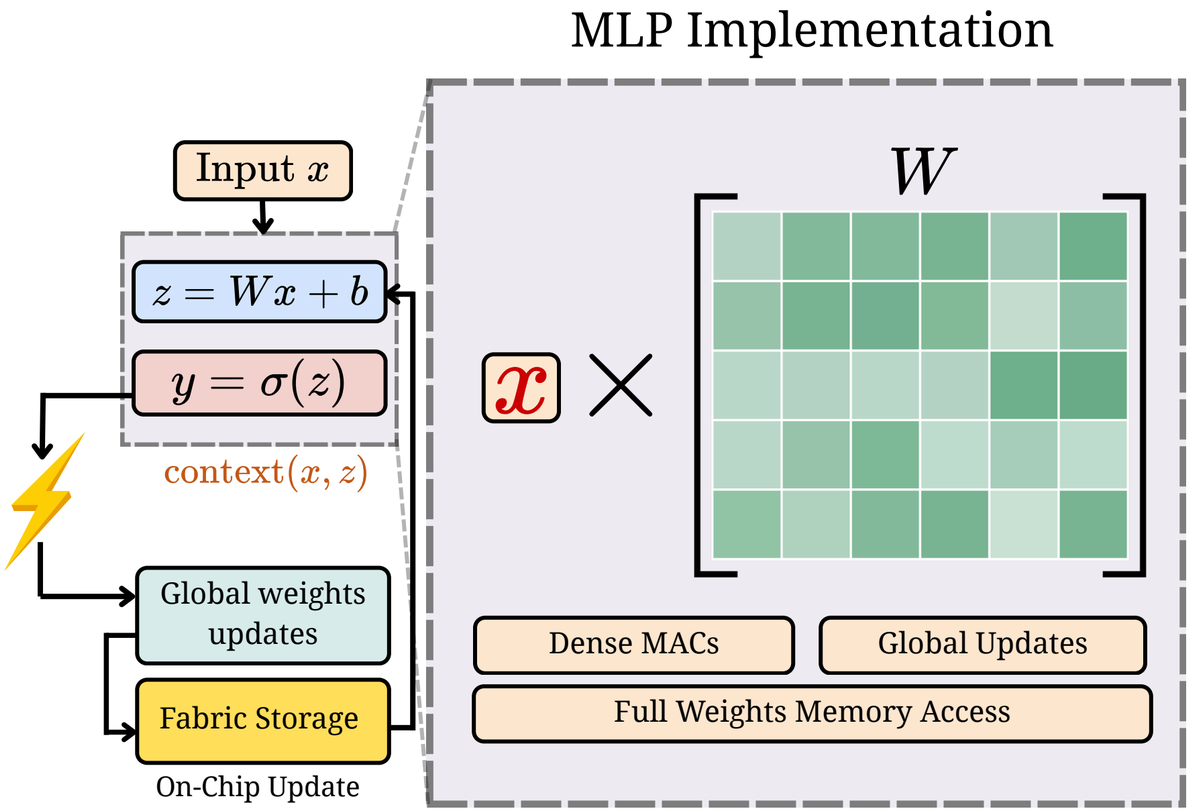
MLP $y_q = \phi\left(\sum_p w_{q,p} x_p\right)$

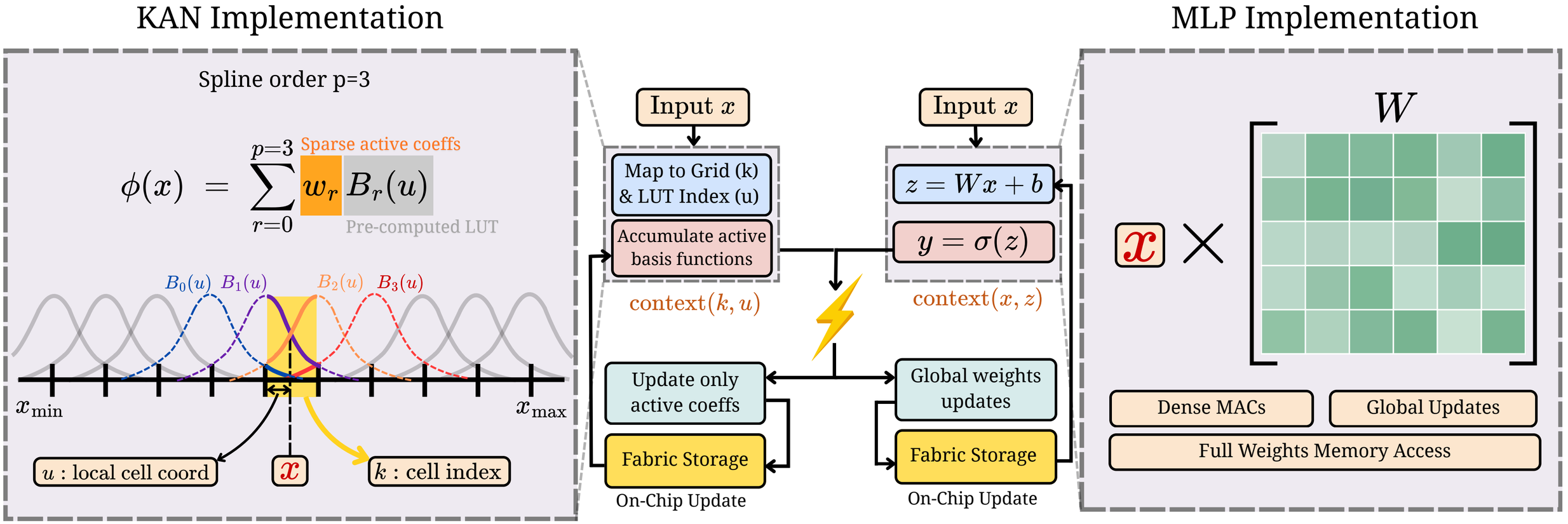
learned activations replace learned weights

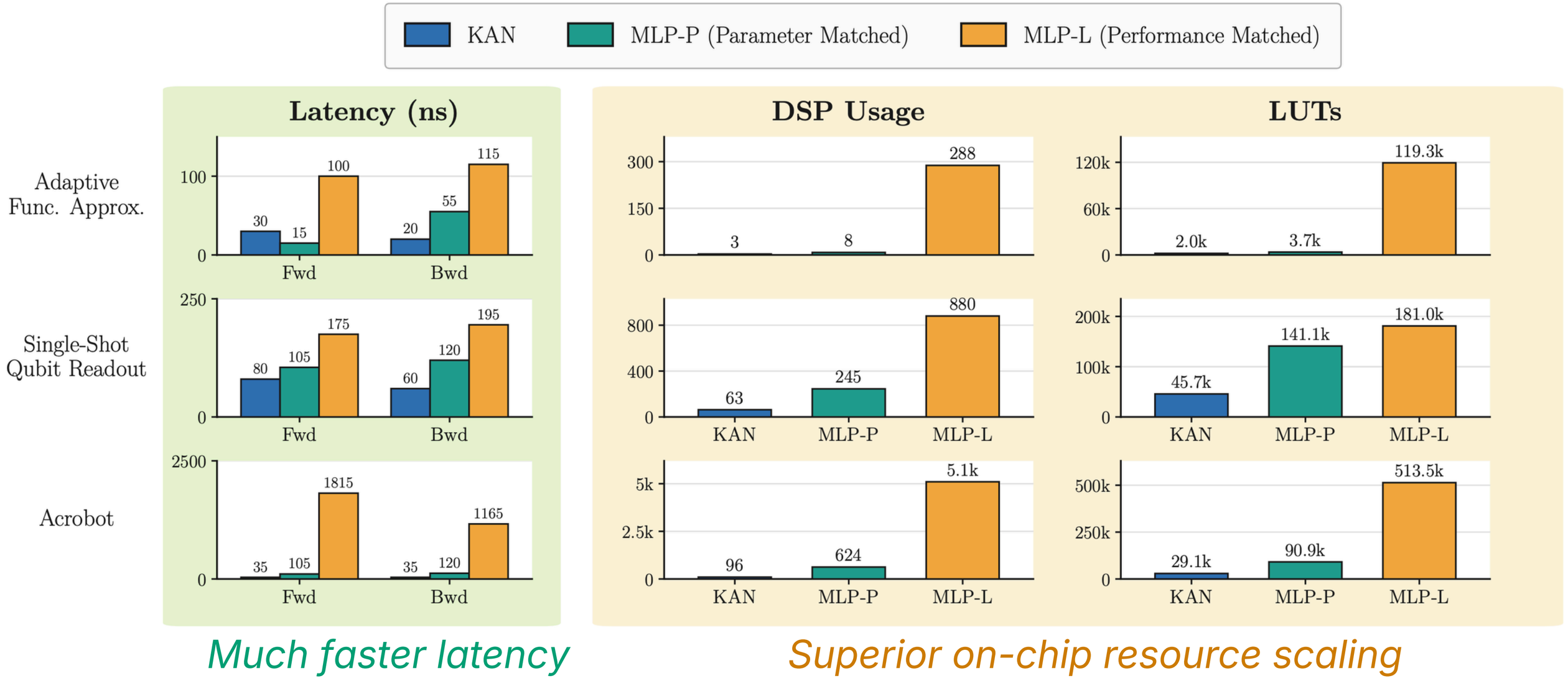
KAN $y_q = \sum_p \phi_{q,p}(x_p)$ where $\phi_{q,p}(x) = \sum_i c_{q,p,i} B_i(x)$

B-splines

- $\{B_i\}$ are ***local*** → small fraction computed per input
- $\{B_i\}$ are bounded → well-behaved under quantization


MLP Implementation
Input x
$z = Wx + b$
$y = \sigma(z)$
$\text{context}(x, z)$
Global weights updates
Fabric Storage
On-Chip Update
x
W
Dense MACs
Global Updates
Full Weights Memory Access


KAN Implementation
Spline order p=3
$\phi(x) = \sum_{r=0}^{p=3} w_r B_r(u)$
Sparse active coeffs
Pre-computed LUT
$B_0(u)$ $B_1(u)$ $B_2(u)$ $B_3(u)$
$x_{\min}$
$x_{\max}$
u : local cell coord
x
k : cell index
Input x
Map to Grid (k) & LUT Index (u)
Accumulate active basis functions
context(k, u)
Update only active coeffs
Fabric Storage
On-Chip Update
Input x
$z = Wx + b$
$y = \sigma(z)$
context(x, z)
Global weights updates
Fabric Storage
On-Chip Update
MLP Implementation
W
x
Dense MACs
Global Updates
Full Weights Memory Access



Much faster latency

Superior on-chip resource scaling

*KAN learners scale to **50k+ params** with **sub-microsecond latencies!***