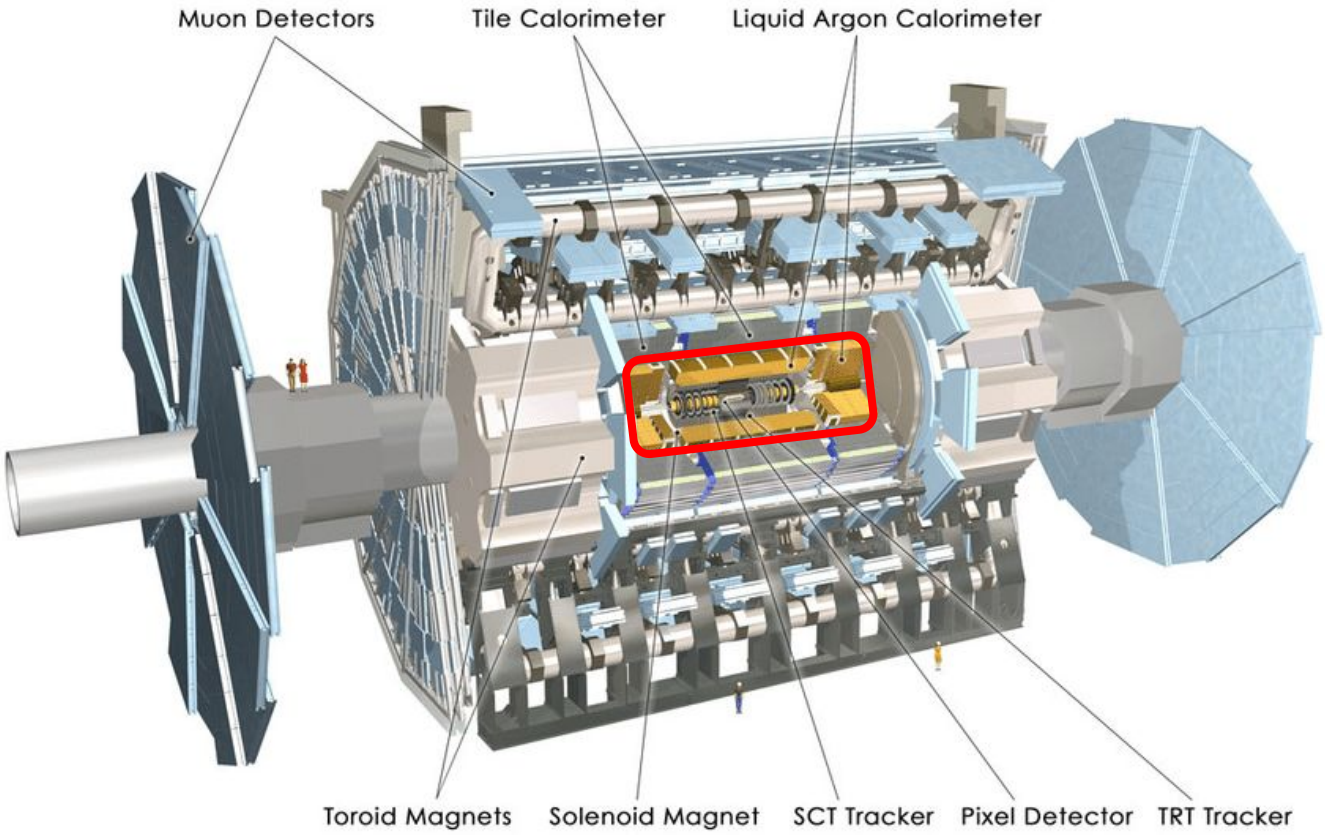
Réseaux de neurones embarqués pour le calcul de l’ énergie déposée dans le calorimètre à argon liquide de l’expérience ATLAS

Georges Aad
CPPM, Aix-Marseille Université, CNRS/IN2P3, Marseille

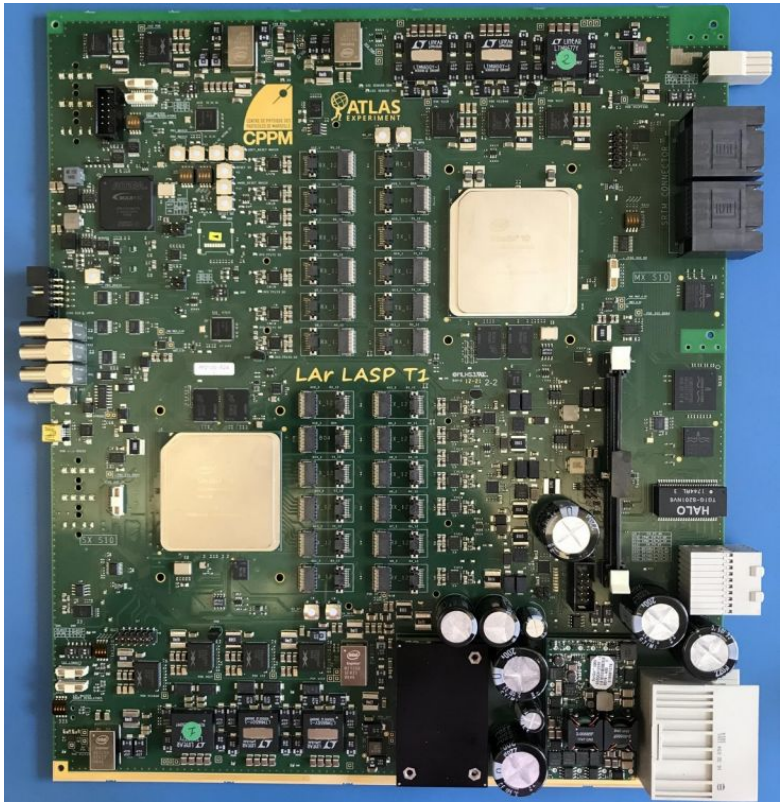
Le calorimètre à argon liquid (LAr) d'ATLAS

- Measures les énergies des particules produites dans les collisions proton-proton au LHC
- Toute l'électronique de lecture de ce calorimètre sera remplacé en 2026-2028
  - Les nouvelles cartes électroniques sont en phase finale de conception



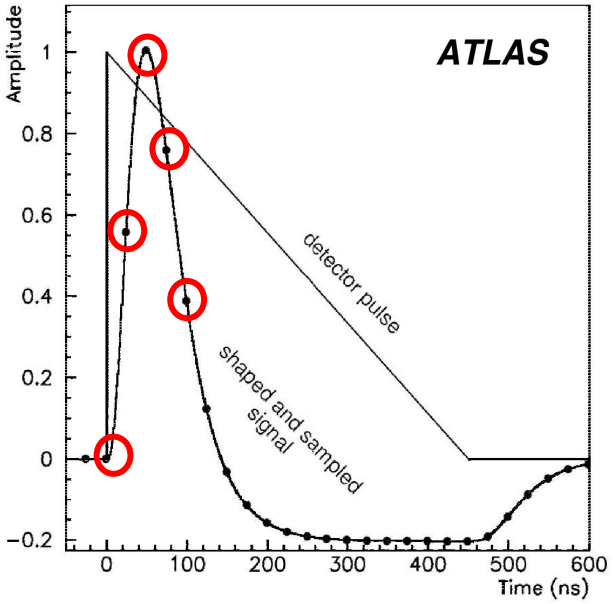
La carte électronique LASP

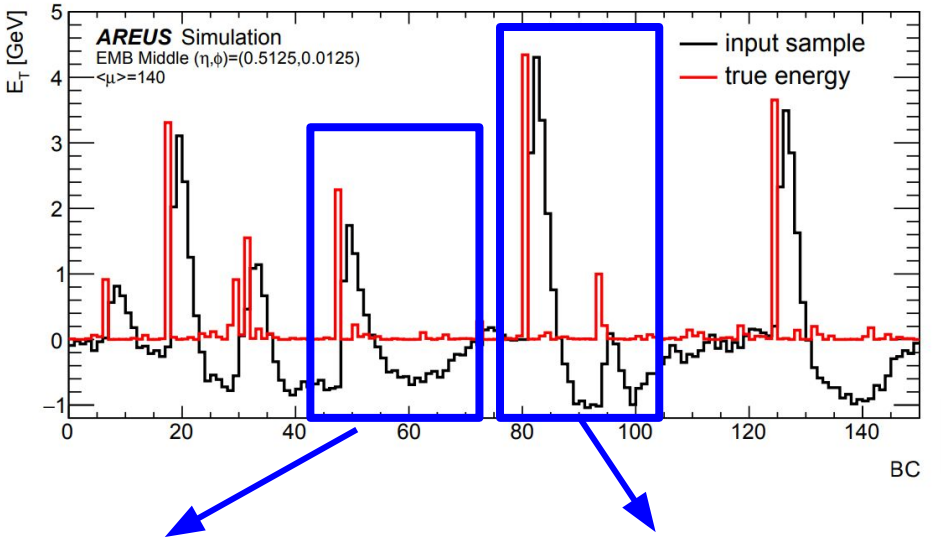
- Carte responsable du traitement des données du calorimètre LAr
  - Calcul à la volée les énergies déposées dans le calorimètre
- Carte contenant 2 FPGAs du type AGILEX
  - Dernière génération des FPGAs d'INTEL
- Carte concu au CPPM
  - Un démonstrateur déjà produit à base de FPGAs Stratix 10
- Un FPGA doit traiter 384 canaux indépendants
  - 300 cartes nécessaires pour tout le calorimètre
- Contraintes importantes sur le firmware
  - Calcul des énergies à 40 MHz (toutes les 25 ns)
  - avec une latence de moins de 125 ns
  - O(1Tb/s) à traiter par carte
  - Nombre d'unité logique limité dans le FPGA



Reconstruction de l'énergie

- Reconstruction de l'énergie à partir du signal électronique du détecteur échantillonné et numérisé à 40 MHz
  - Filtrage optimal pour calculer l'amplitude du signal (∝ energie)
- Les performances de OFMax se dégrade significativement dans le cas de signaux qui se chevauchent





Signal isolé

Signaux que se chevauchent
Distorsion de la forme du signal

Energy from Optimal-Filter (OF)

n = 5 in this talk

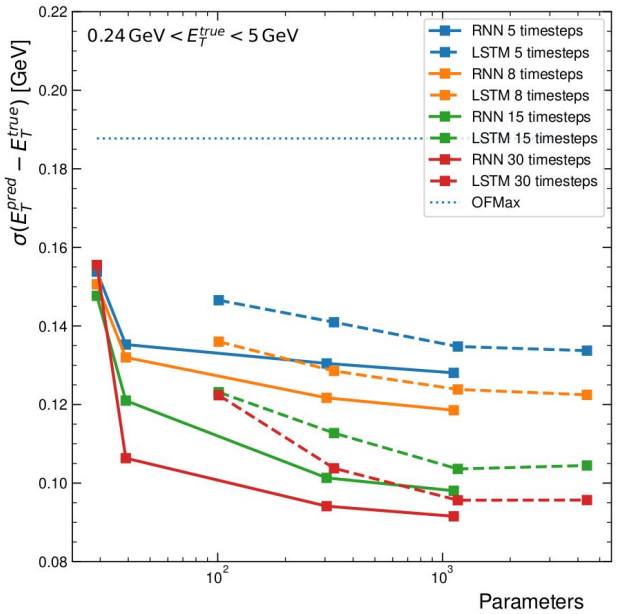
$$E(t) = \sum_{i=t}^{t+n} a_i \cdot s_i$$

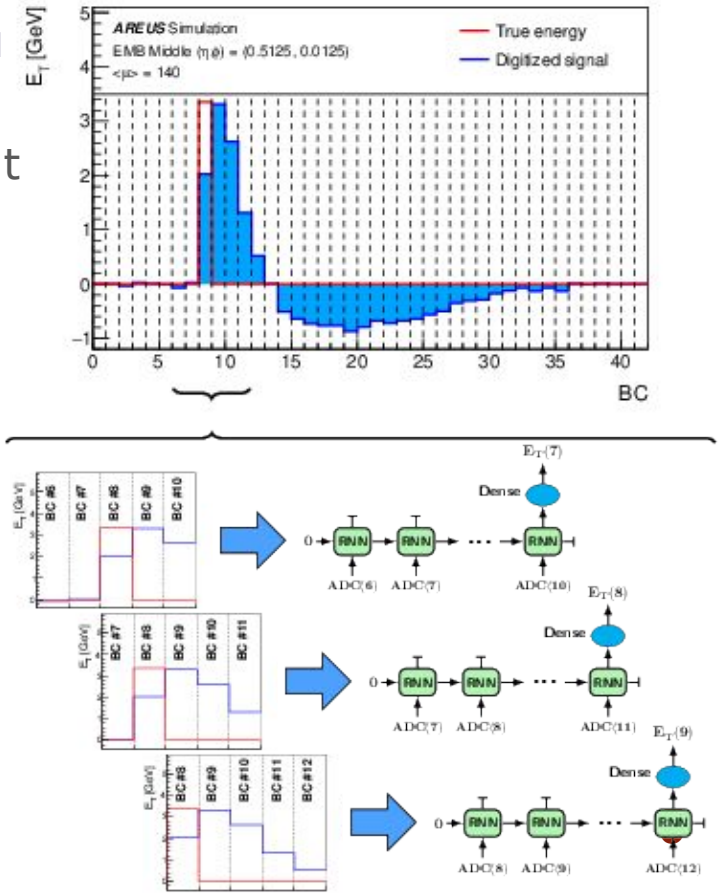
Pulse Samples

Pre-set coefficients (fit of the peak)

Reconstruction de l'énergie avec des RNNs

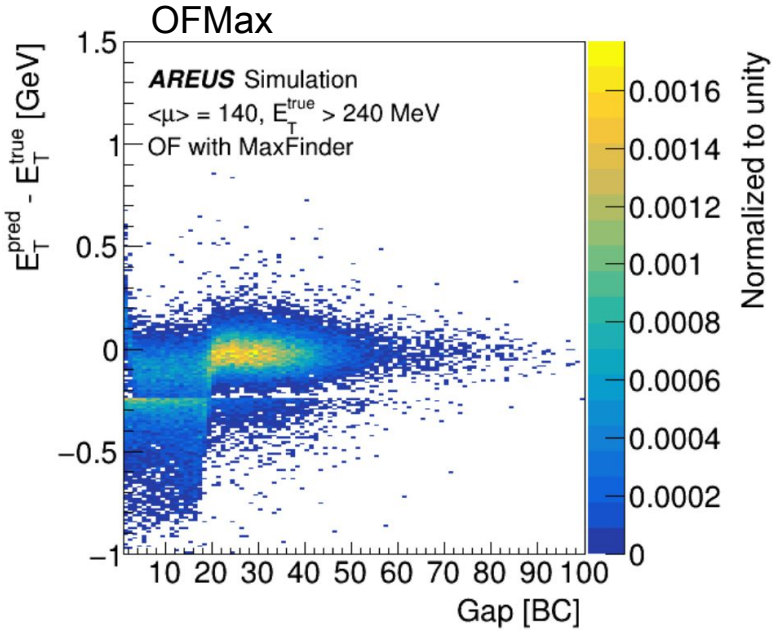
- On se concentre sur les réseaux de neurones récurrents (RNN)
  - Trois types of RNNs: vanilla-RNN, GRU et LSTM
- Une chaîne d'échantillons est utilisée comme entrée pour les RNNs
  - 4 échantillons sur le pic pour calculer l'amplitude
  - Plusieurs échantillons dans le passé pour corriger les effets des événements passés
- Optimisation des paramètres et architecture du réseaux pour la meilleur resolution en energie
  - Préférence pour les petits réseaux qui peuvent rentrer dans le FPGA

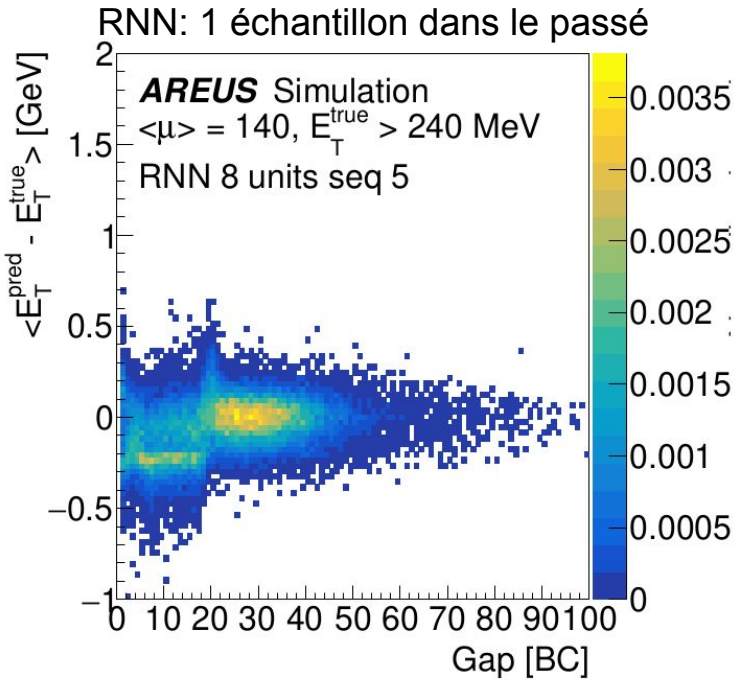


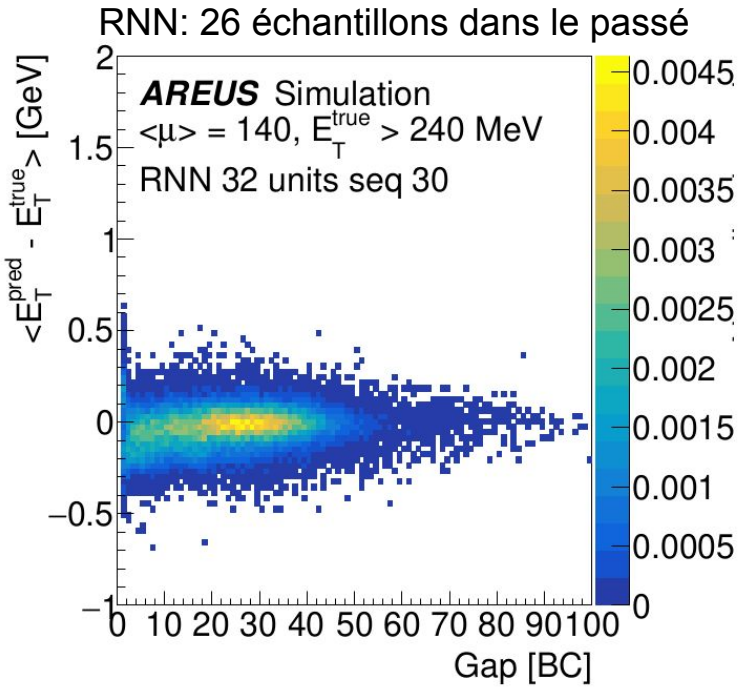


Performance des RNNs

- Etude des performances en fonction du temps écoulé entre deux dépôts d'énergie (GAP)
  - Dégradation des performances à petit GAP
- Les RNNs corrigent les dégradations à petit GAP en utilisant les infos dans le passé







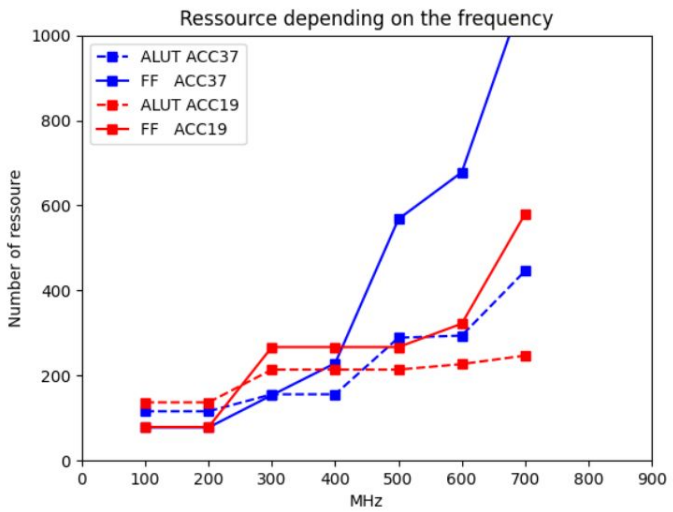
Implementation firmware en HLS

- Implémentation du Vanilla RNN initialement en HLS dans un Stratix 10
- Plusieurs optimisations nécessaires pour que les RNNs entre dans le FPGA
  - Opération avec nombre à virgule fix (pas flottante comme dans le software)
  - Optimisation du nombre de bits pour représenter les nombres
  - Optimisation de la façon d'arrondir les résultats des opérations arithmétiques
  - Optimisation de l'implémentation des multiplications matricielle
  - Time Multiplexing: Plusieurs canaux traités par une seule instance de RNN

- Implémentation des multiplications matricielles
  - C++ Naive: INTEL HLS optimize le tout
  - ACC37: accumulation dans un enchaînements DSPs
  - ACC19: accumulation dans les ALUT
- L'implémentation optimale dépend fortement de la fréquence de traitement

$$A.B = \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_8 \end{bmatrix} \cdot \begin{bmatrix} b_0 \\ b_1 \\ \vdots \\ b_8 \end{bmatrix} = \sum_{i=0}^{7} a_i.b_i$$

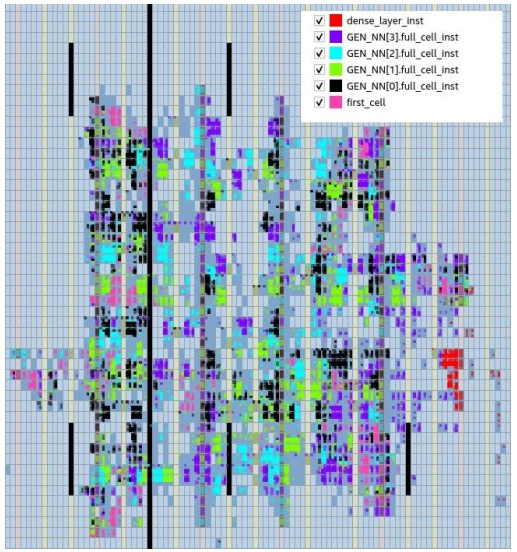
	Implementation	ALUTs	FF	DSP
@100 MHz	C++ style	709	222	8
	ACC37	116	79	4
	ACC19	137	78	4



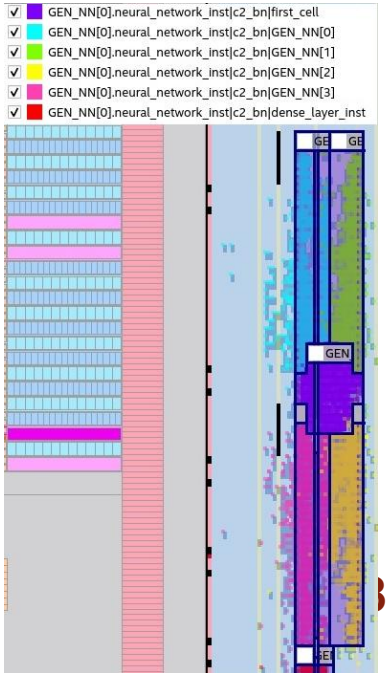
VHDL implementation of Vanilla RNN

- Même avec les optimisations, l'implémentation en HLS ne réponds pas au cahier de charge
- Passage en VHDL nécessaire avec deux optimisations:
  - Placement forcé dans le FPGA
    - Permet un meilleur control sur les violations de timing dans le FPGA
  - Utilisation de la compilation incrémentale
    - Garder les instances de RNN qui passe les contrainte de temps et recompiler le reste

HLS placement



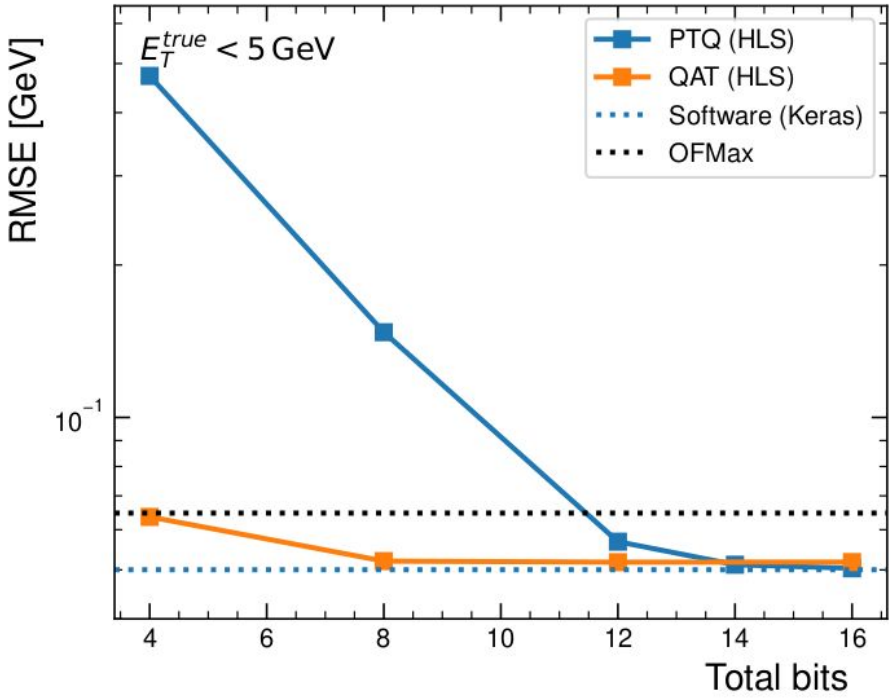
VHDL forced placement



Vanilla RNN de dimension 8 avec 5 échantillons	N networks x multiplexing	ALM	DSP	FMax	latency
target	**384 channels**	**30%**	**70%**	**-**	**125 ns**
HLS (no multiplexing)	384x1	226%	529%	-	322 ns
HLS optimized	37x10	90%	100%	393 MHz	278 ns
VHDL optimized	28x14	18%	66%	561 MHz	116 ns

Quantification des operations arithmetiques

- Le passage d'opérations arithmétiques en virgule flottante à virgule fix engendre une perte de précision
  - Augmentation du nombre de bits nécessaire dans le FPGA
    - 18 bits nécessaire pour avoir une perte de précision de moins de 0.1%
- Ce problème peut être évité en utilisant les virgules fix directement durant l'entraînement du réseau (possible avec qKeras)
  - Permet de diminuer d'un facteur 2 le nombre de bit nécessaire pour le calcule dans le FPGA



Conclusion

- Les RNNs permettent de calculer les énergies dans le calorimètre LAR d'ATLAS avec une meilleur précision que les algorithms actuelles
- Plusieurs architecture sont testées et le travail d'optimisation continu

- On a réussi à implémenter un Vanilla-RNN (de dimension 8 et 5 échantillons) dans un Stratix 10 en respectant le cahier de charge très restreint d'ATLAS
- Amélioration nécessaire pour pouvoir implémenter des RNNs plus grand et plus performant dans le firmware
  - Mais bientôt on aura un plus gros FPGA (Agilex) qui a plus de puissance de calcul

- Deux papiers et une thèse publiés:
  - *Comput Softw Big Sci* **5**, 19 (2021)
  - JINST 18 (2023) P05017
  - *Etienne Fortin, PhD thesis (2022)*