ML-assisted SRAM Soft Error Rate Characterization: Opportunities and Challenges

Masanori Hashimoto, Ryuichi Yasuda, Kazusa Takami, Yuibi Gomi

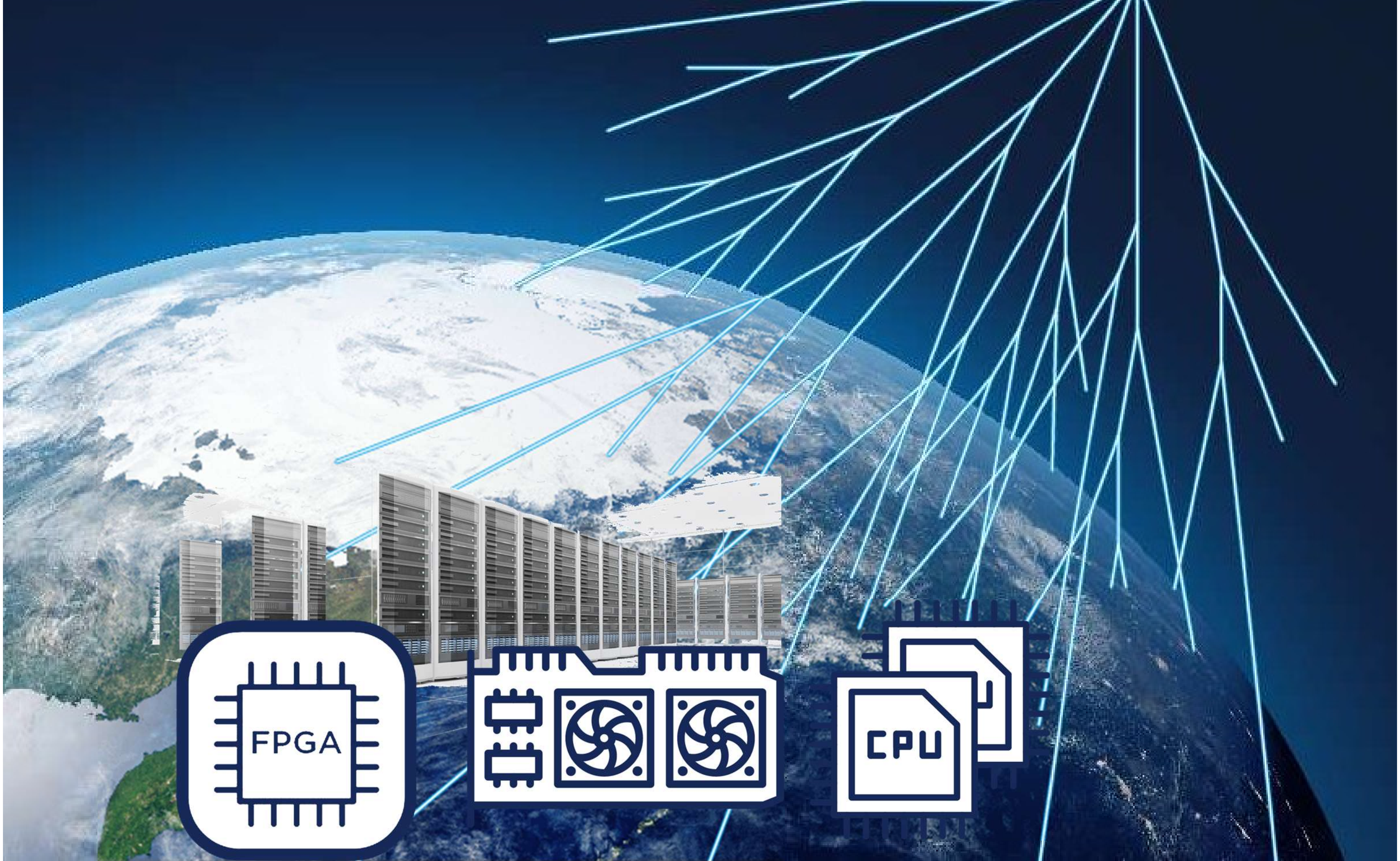
Dept. Informatics, Kyoto University

Kozo Takeuchi

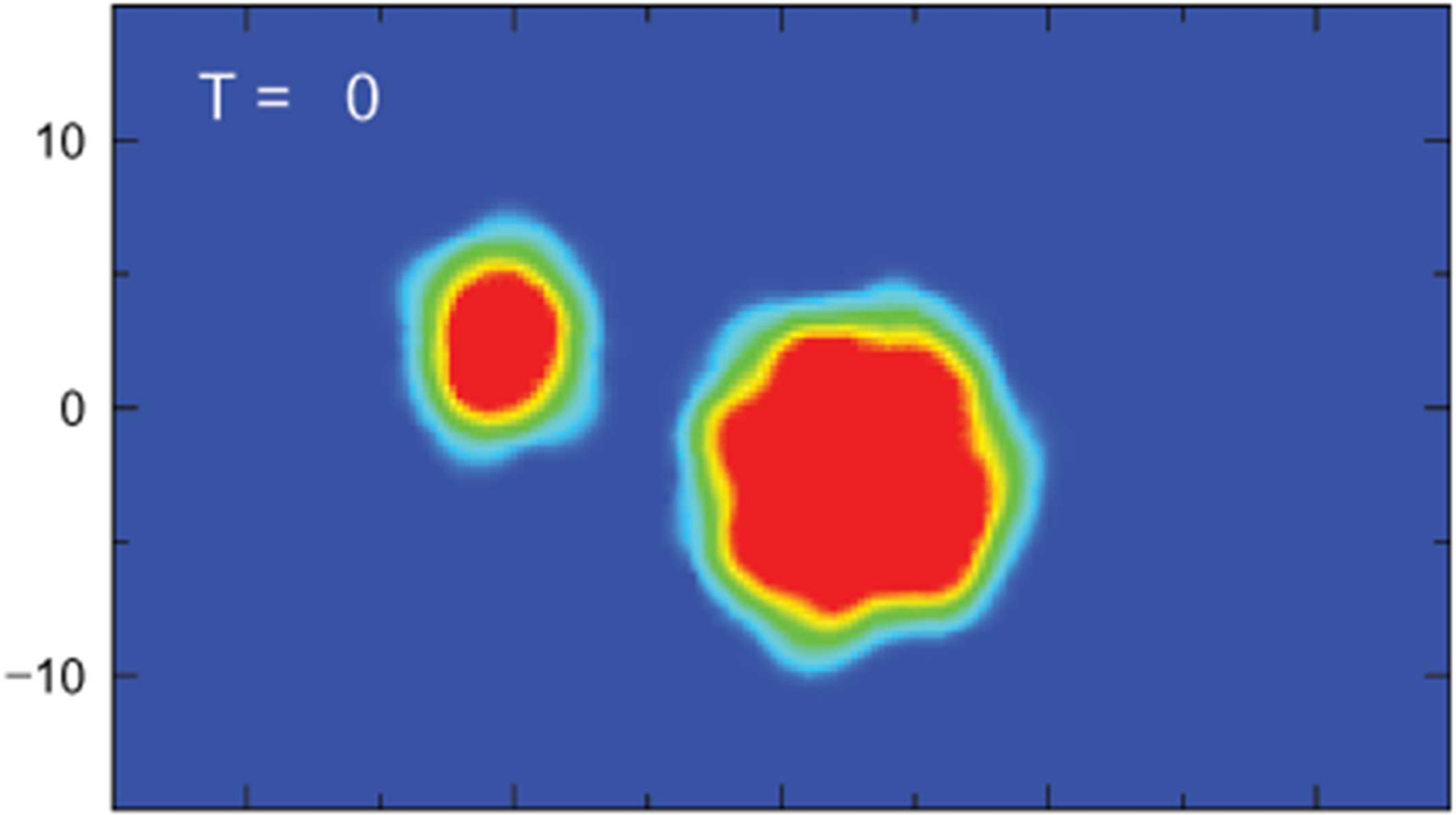
JAXA

hashimoto@i.kyoto-u.ac.jp

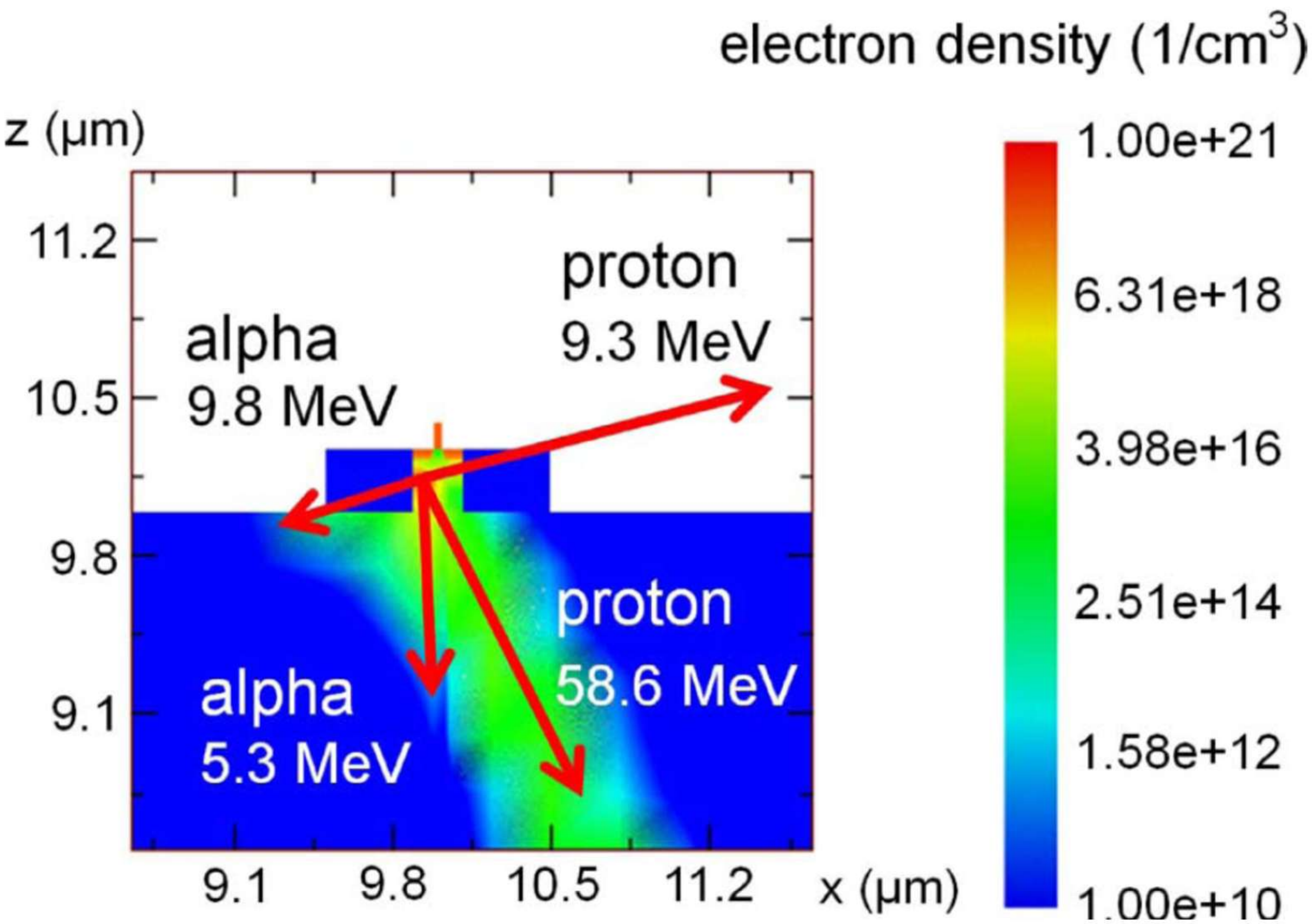
Cosmic ray-induced neutrons and muons are falling into VLSI chips



Example of nuclear reaction



Example of reaction in VLSI chip

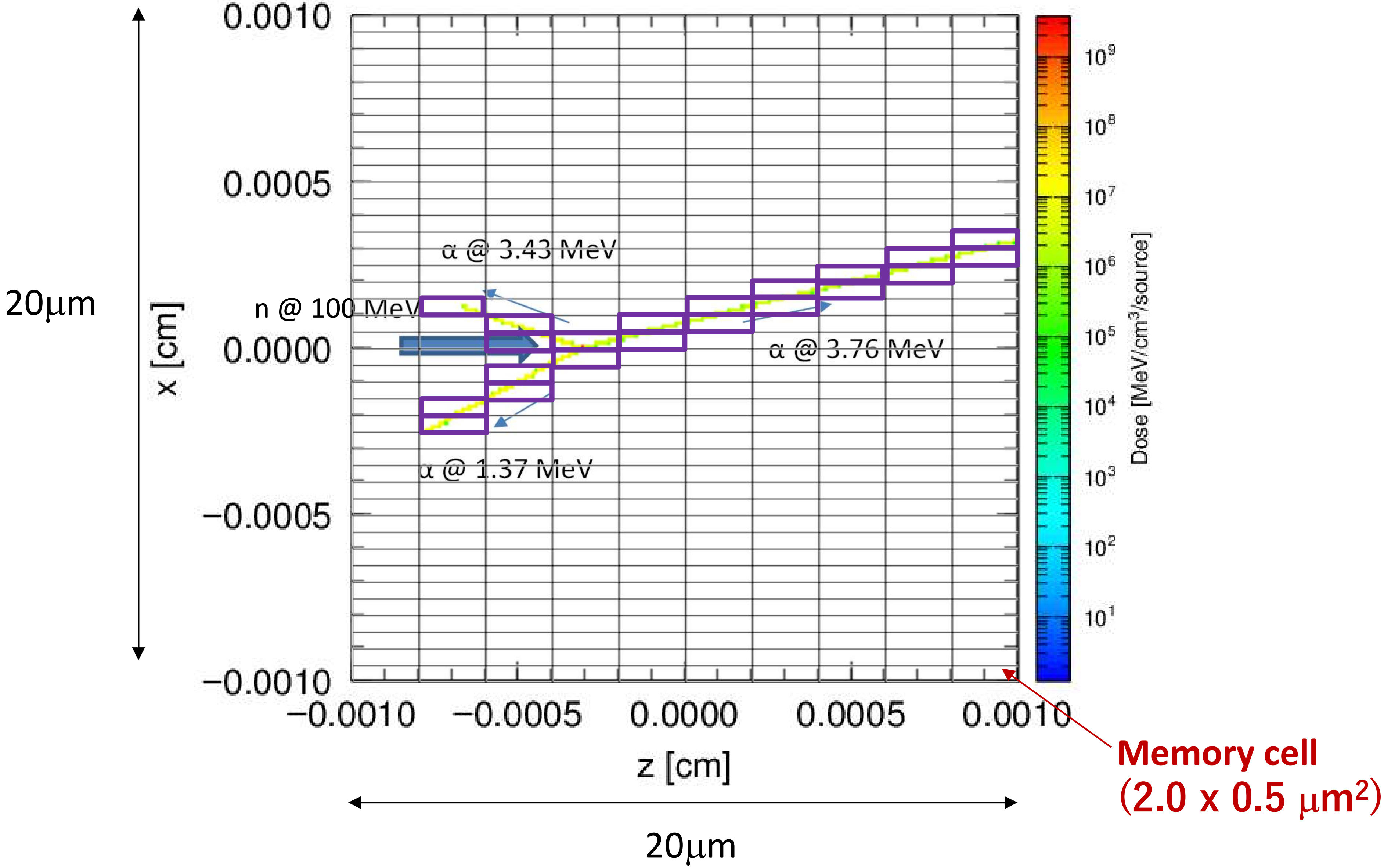


▶ **Injected charge may result in bit flip called soft error.**

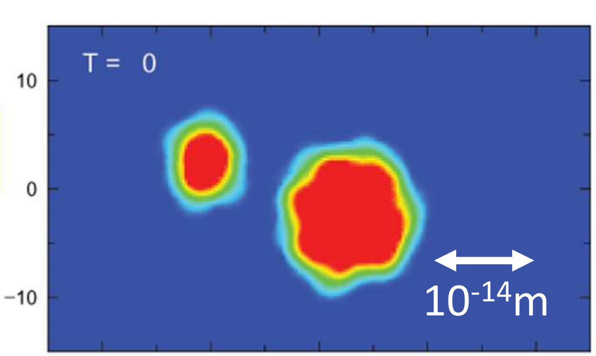
Fig. 3. Cross section view of the initial charge distribution by the nuclear reaction, $n + {}^{28}Si \rightarrow 3n + 2p + 2\alpha + {}^{16}O$, at the incident energy of 233 MeV. The track of ${}^{16}O$ ion with its kinetic energy of 4.23 MeV is not depicted because the direction of motion is nearly parallel to the y-axis.

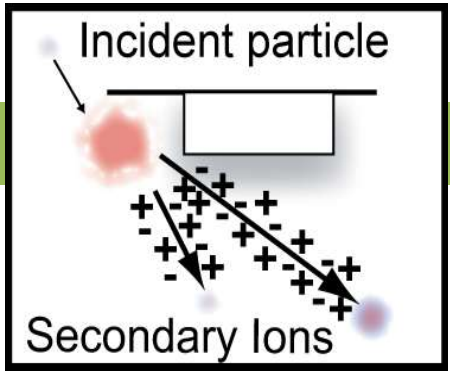
[1] S. Abe, et. al, "Multi-scale Monte Carlo simulation of soft errors using PHITS-HyENEXSS code system," *IEEE Trans. Nuclear Science*, 2012

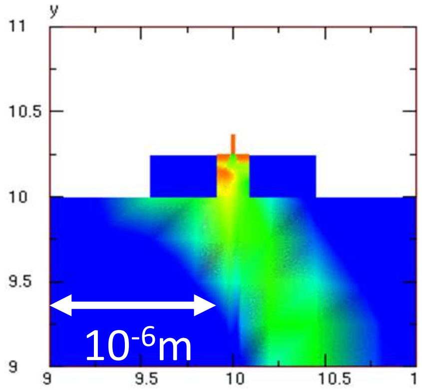
Example @ 65nm

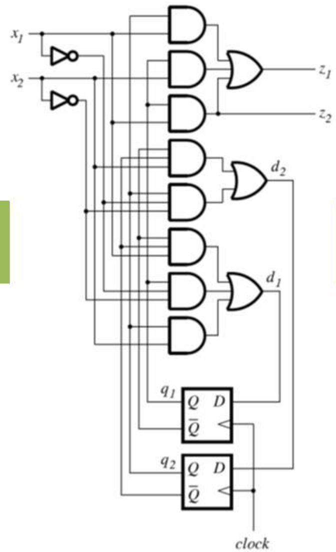


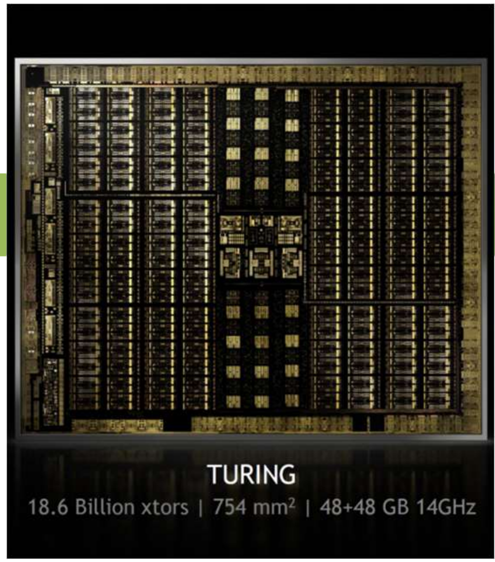
Multi-physics multi-layer phenomena with diverse temporal and spatial scales





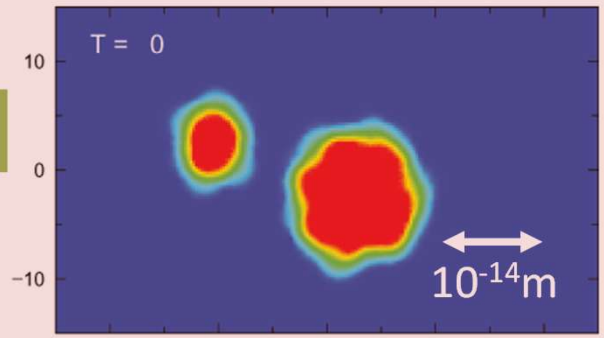


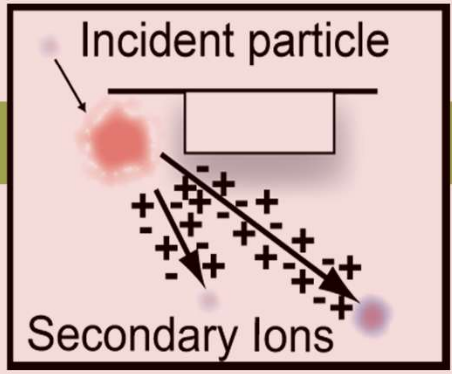


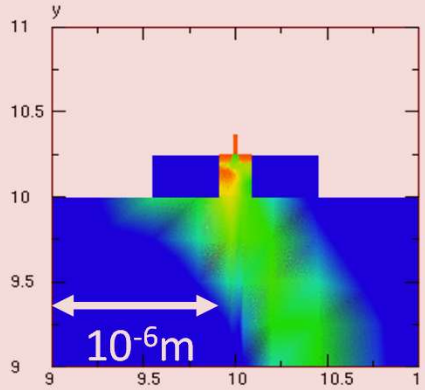


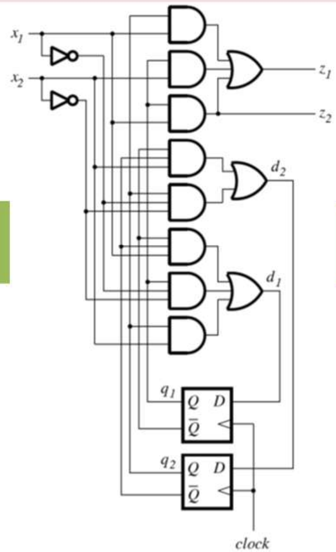
Multi-physics multi-layer phenomena with diverse temporal and spatial scales

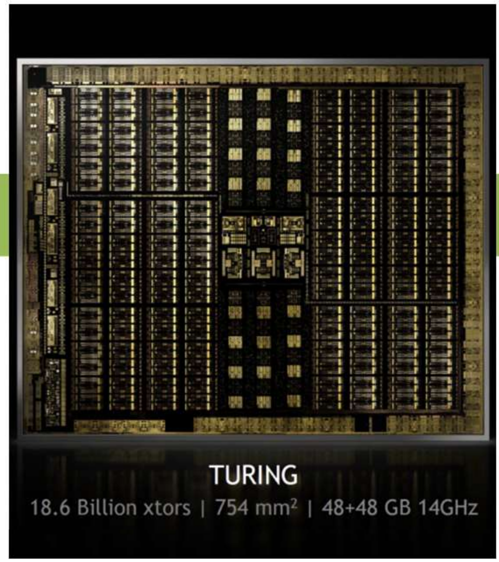
SRAM Soft Error Rate Characterization (Focus of this talk)



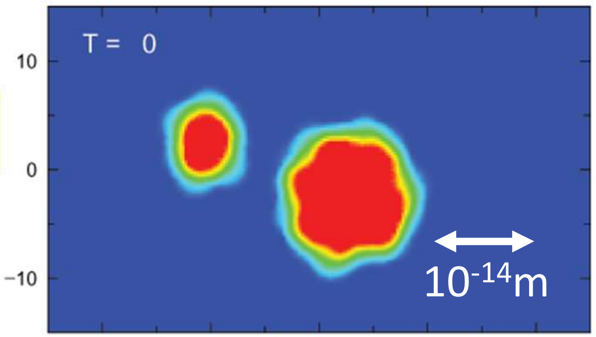


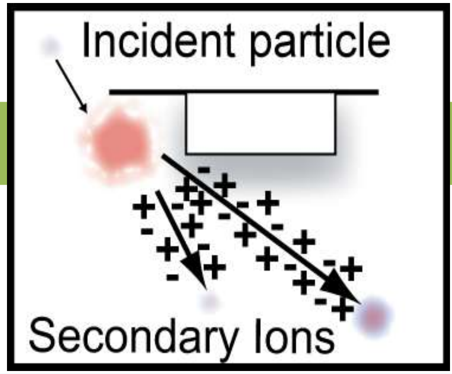


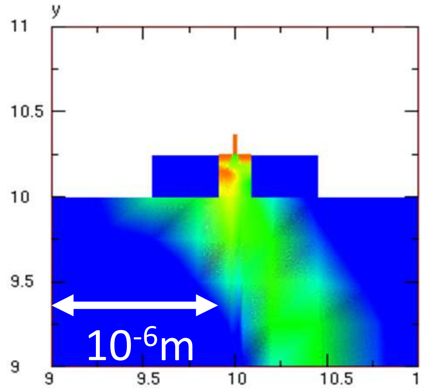




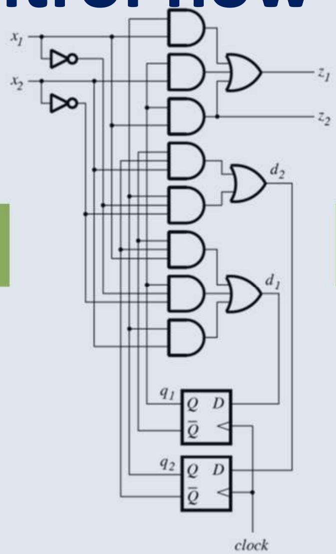
Multi-physics multi-layer phenomena with diverse temporal and spatial scales

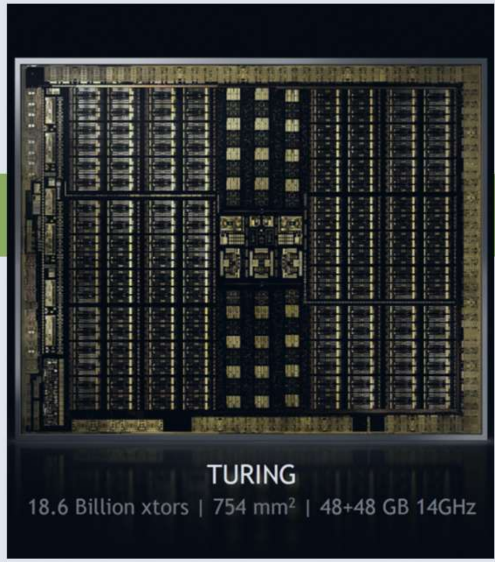






Control flow monitoring (6D-5, Wed.)





Agenda

- Background: soft error
- **Conventional soft error rate (SER) simulation and its challenges**
- Proposed method and experiments
- Future directions and conclusions

Conventional SER simulation

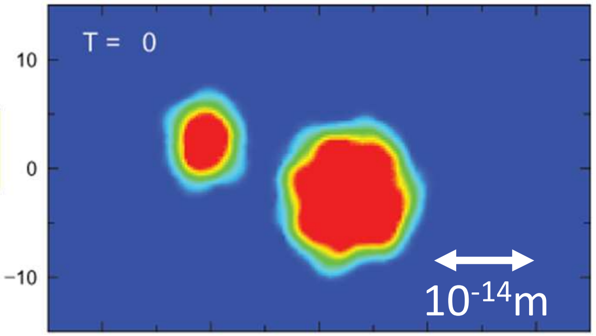
Monte Carlo simulation aiming to reproduce nuclear physics, charge deposition and SRAM cell behavior.

Simulators: PHITS[2], Geant4 [3], etc

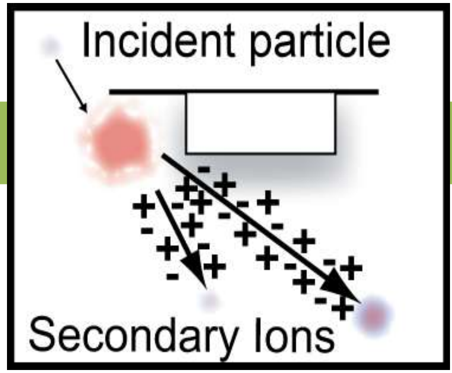
[2] T. Sato, et al., “Recent improvements of the particle and heavy iontransport code system – PHITS version 3.33,” *J. Nuclear Sci. & Tech.*, 2024.
[3] S. Agostinelli, et al., “Geant4 - a simulation toolkit, ” *Nuclear Instruments and Methods in Physics Res. Sec. A*, 2003.

A number of event data are generated.
(particle type, energy, location, direction, resultant charge deposition)

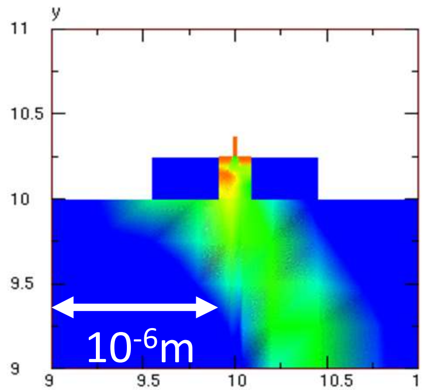
(1) Nuclear physics



(2) Charge deposition



(3) SRAM Cell behavior



Conventional SER simulation

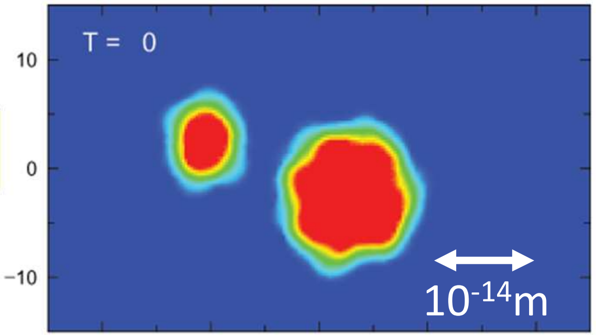
Monte Carlo simulation aiming to reproduce nuclear physics, charge deposition and SRAM cell behavior.

Simulators: PHITS, Geant 4, etc

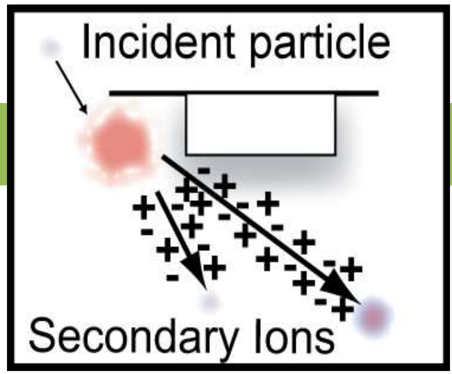
A number of event data are generated.
(particle type, energy, location, direction, resultant charge deposition)

For each event, how can we know if an upset occurs or not?

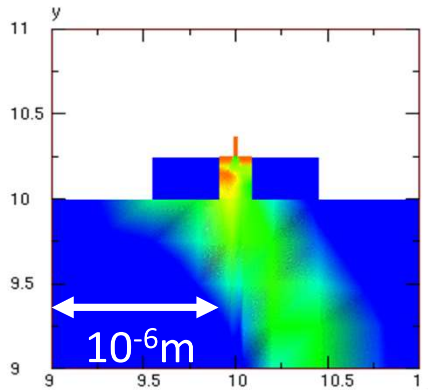
(1) Nuclear physics



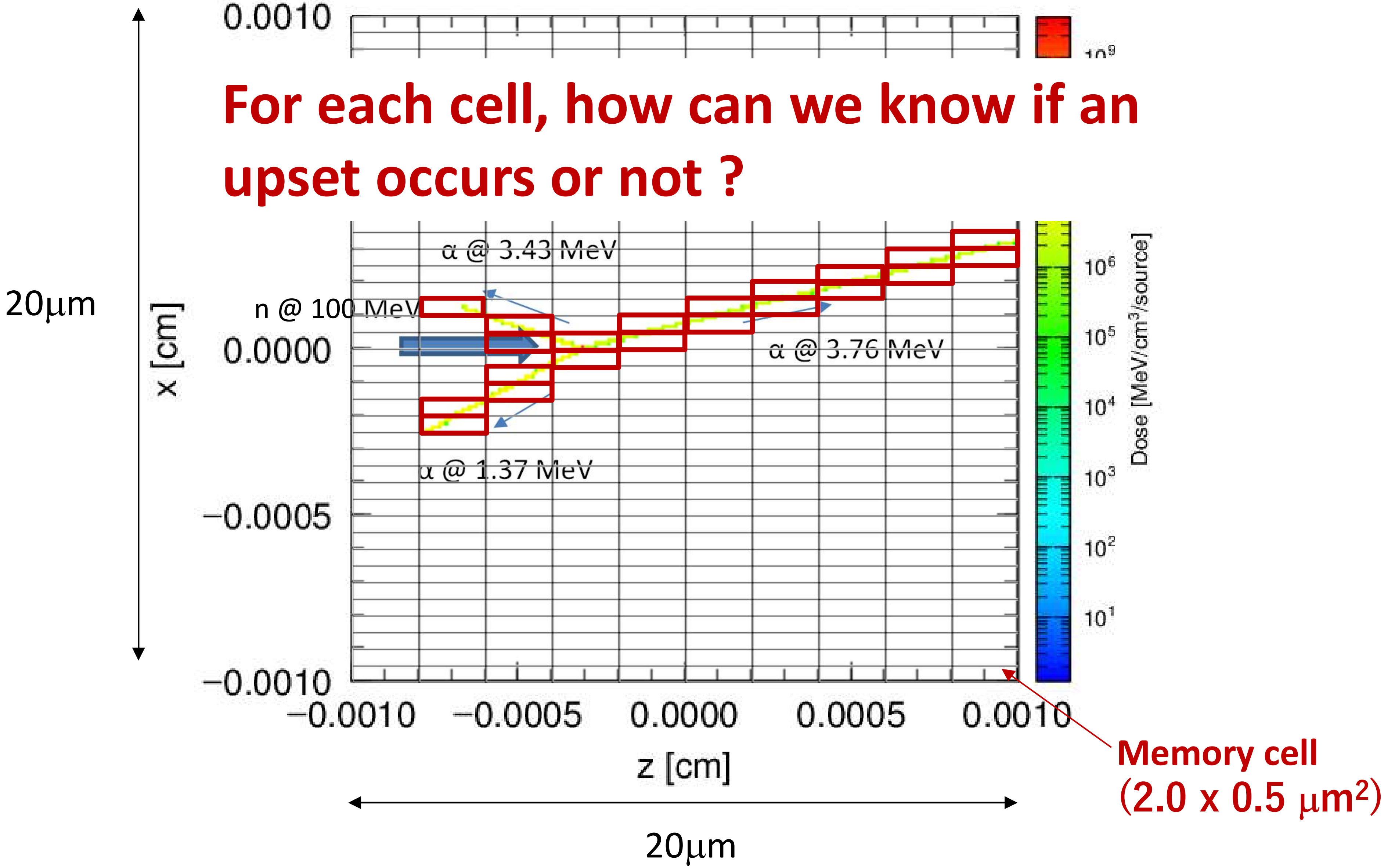
(2) Charge deposition



(3) SRAM Cell behavior

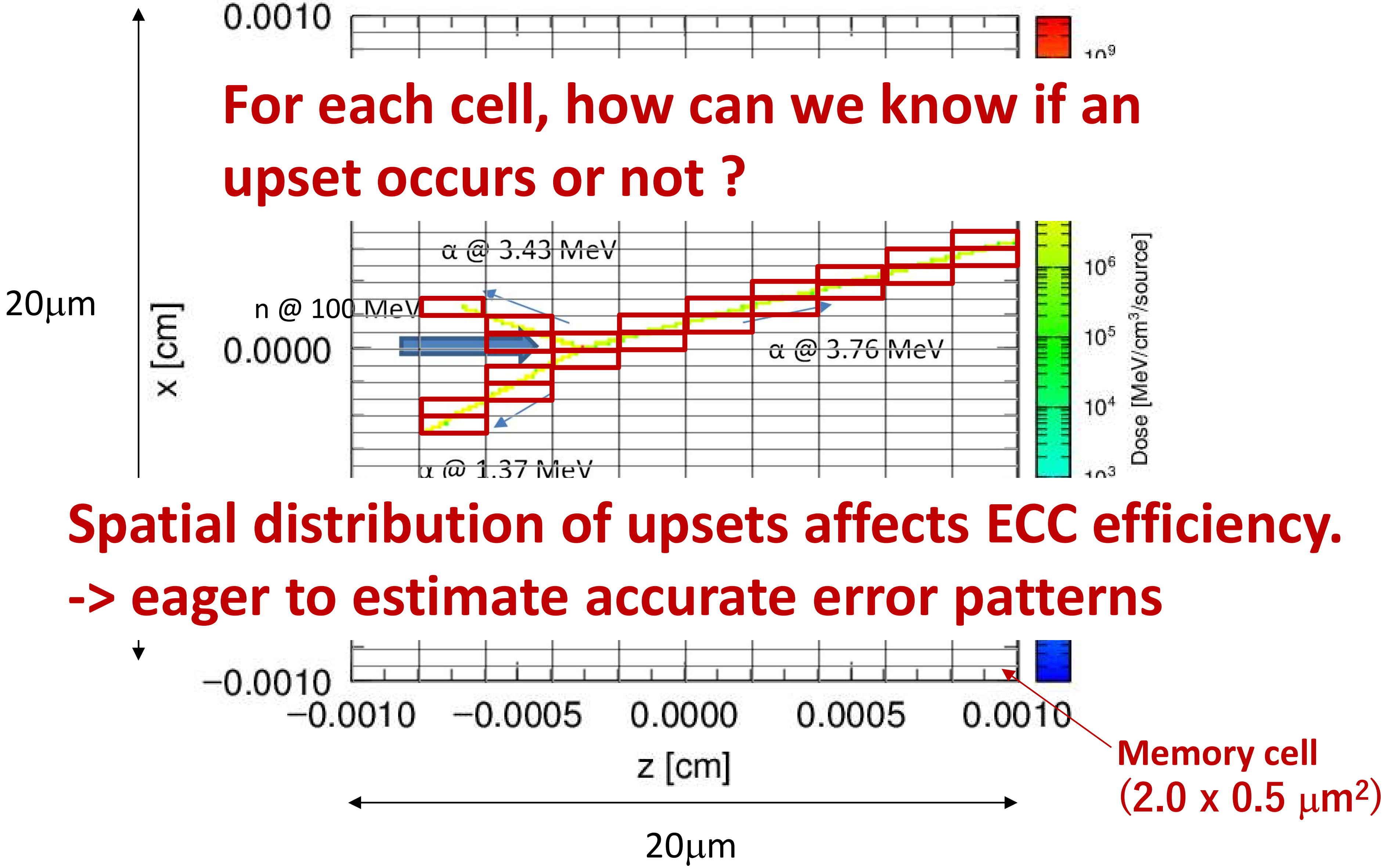


Example @ 65nm



For each cell, how can we know if an upset occurs or not ?

Example @ 65nm

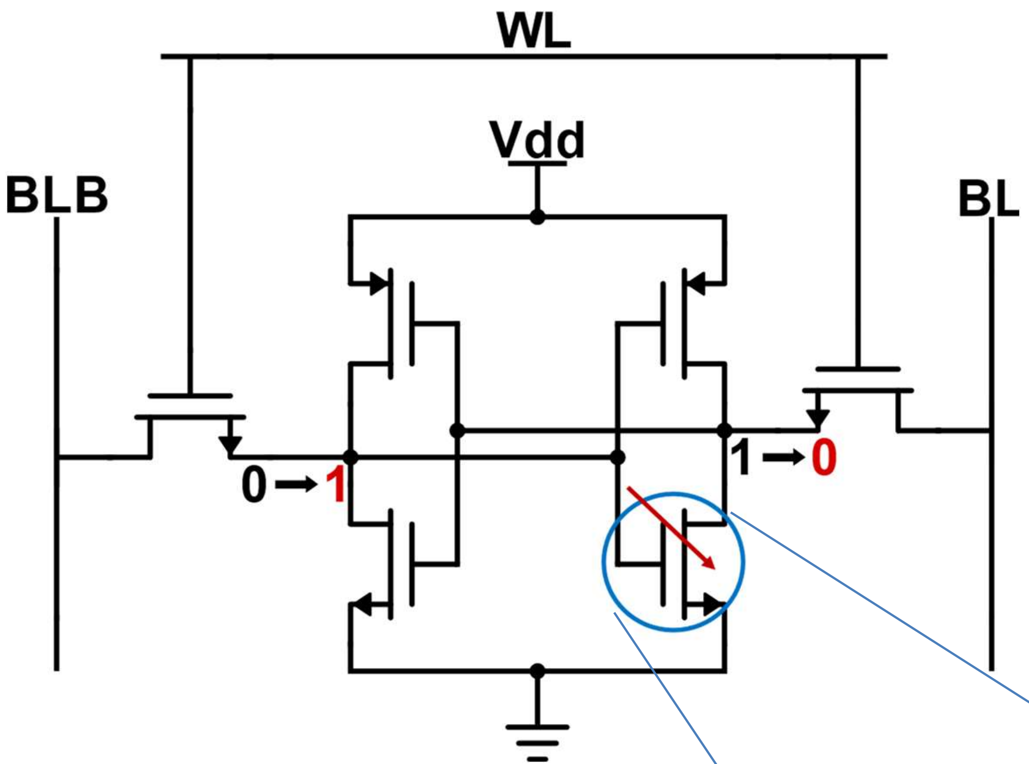


For each cell, how can we know if an upset occurs or not ?

Spatial distribution of upsets affects ECC efficiency.

-> eager to estimate accurate error patterns

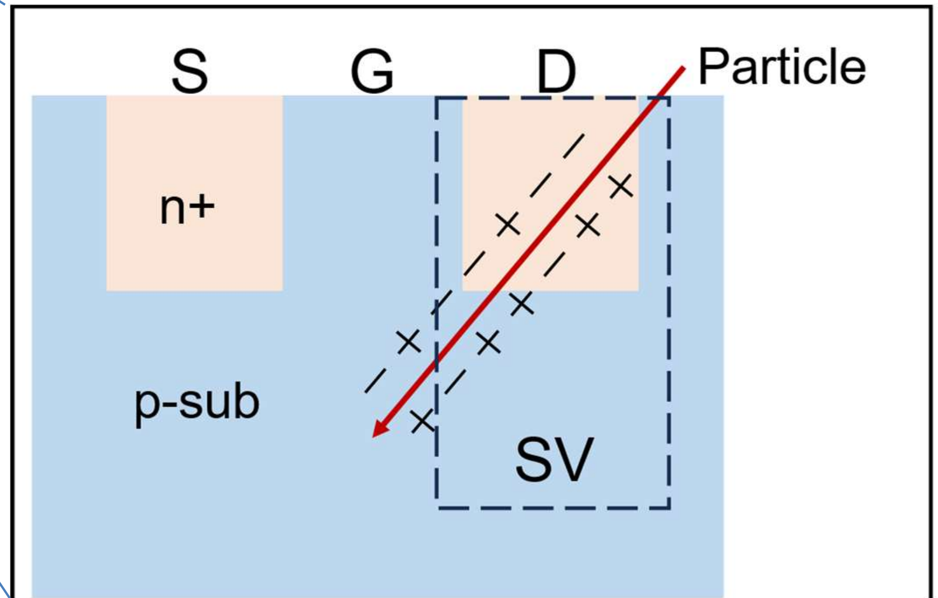
Conventional sensitive volume (SV) method



Is the charge collected to drain (Q_{coll}) large enough?

Q_{coll} $(=Q_{dep}*C_{eff}) > Q_{crit}$?

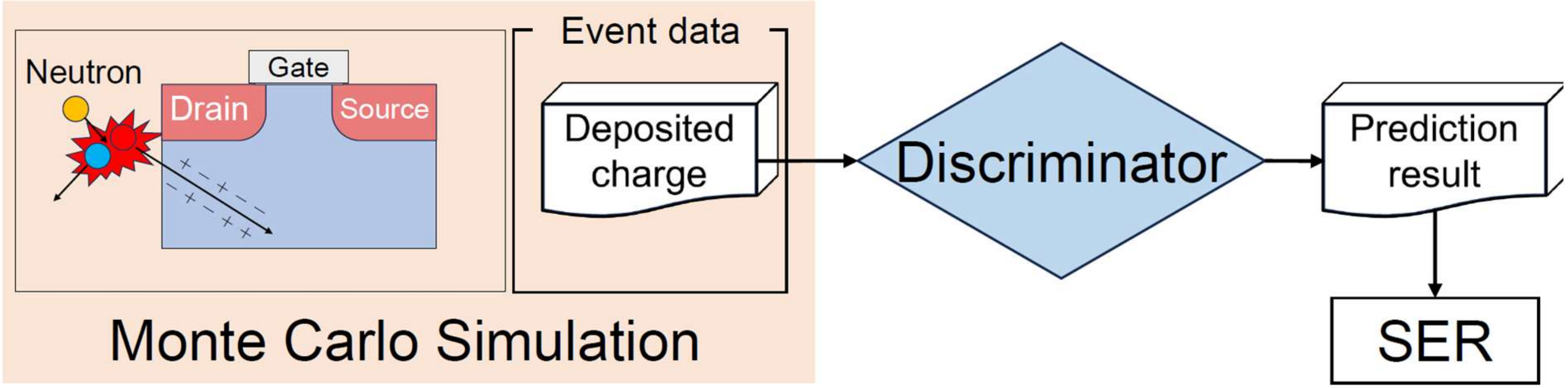
Q_{dep}: charge deposited in SV
C_{eff}: charge collection efficiency
Q_{crit}: critical charge (threshold)



Pro: very fast discrimination (upset or not)

Con: determining SV, Q_{crit}, C_{eff} is difficult
low event-wise accuracy

For better discriminator accuracy



- Assign multiple SVs (MSV) to consider spatial C_{eff} difference [4]
  - Fast, yet not sufficiently accurate.
- Give each charge deposition to TCAD and perform transient simulation [5]
  - Very accurate, but prohibitively slow
- Construct and use ML-based classifiers [6, 7]
  - Validated for 65nm SOI only
    - Not validated for recent FinFET SRAMs
  - Prepared training data using TCAD
    - Suffering from long TCAD simulations

[4] K.M. Warren, et al., *IEEE Trans. Nucl. Sci.*, 2007. [5] S. Abe et al., *IEEE Trans. Nucl. Sci.*, 2012.
[6] S. Hirokawa, et al., *RADECS*, 2016. [7] M. Hashimoto, *SISPAD*, 2017.

Contribution of this work

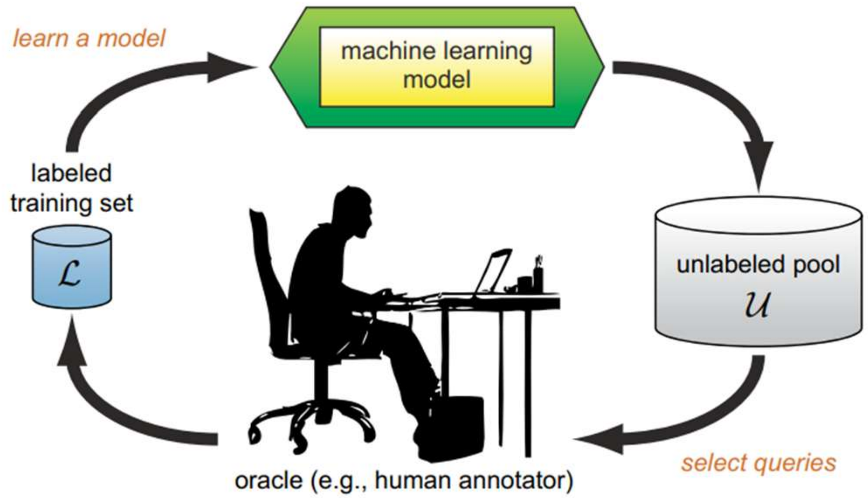
- Propose an active learning-based classifier construction method w/ less TCAD simulations
- Validate it with 12nm FinFET SRAM
- Discuss future directions

Agenda

- Background: soft error
- Conventional soft error rate (SER) simulation and its challenges
- **Proposed method and experiments**
- Future directions and conclusions

Active learning (AL)

- Identifies unlabeled samples which are most likely to improve the model if labeled
- Repeats the process of labeling, retraining and unlabeled sample selection
- Pros
  - – Reduced labeling costs
  - – Efficient esp. when labeling is expensive
  - – Faster accuracy improvements

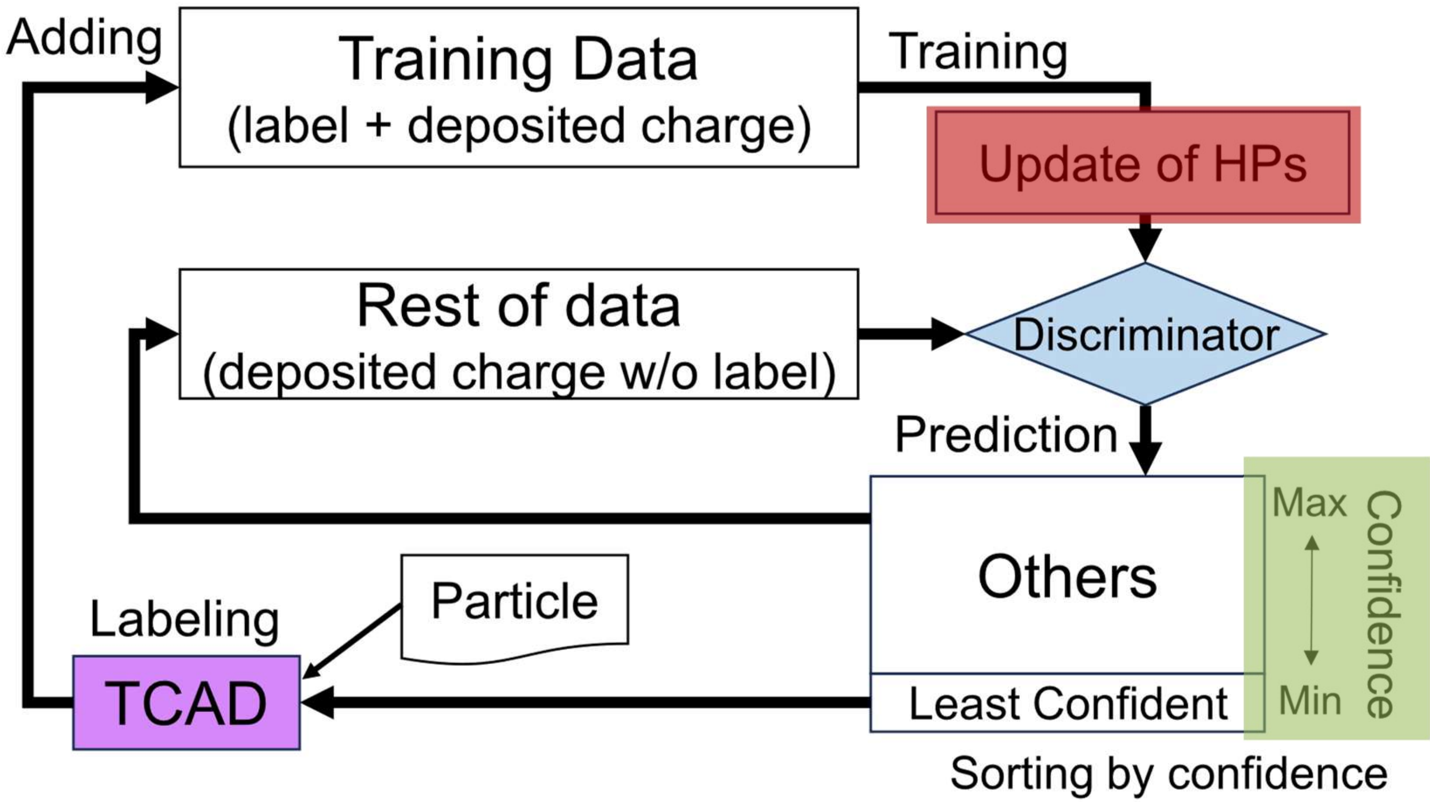


TCAD in our case

[8] B. Settles, "Active Learning," Springer, 2009.

Flow of AL-based training

HP: hyperparameter

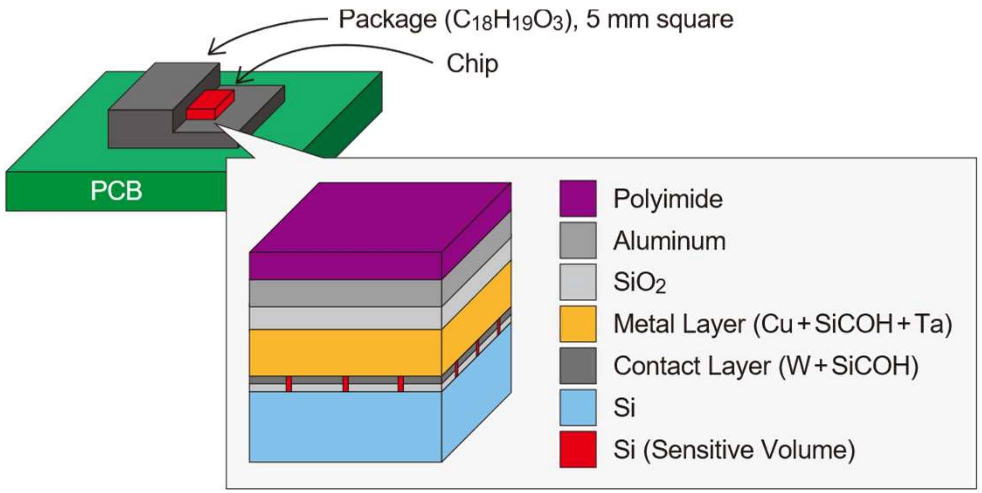


As # of labeled samples increases, more complex models can be trained for higher accuracy.

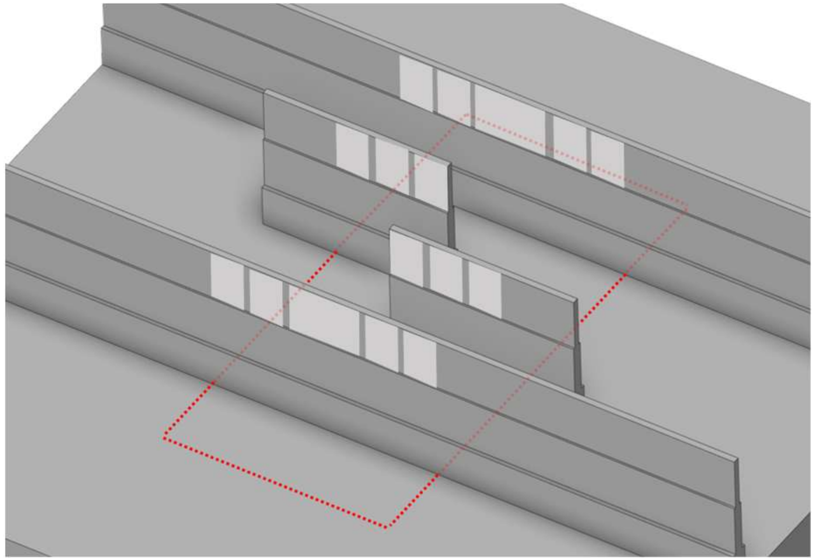
Sorted by |*p*-0.5|
***p*: estimated upset probability**

Simulation setup

- DUT: 12nm 6T-SRAM (Vdd=0.8V)
- Physics simulation (nuclear reaction and particle transport): PHITS [2]
- TCAD (device simulation) : HyENEXSS [9]



PHITS model

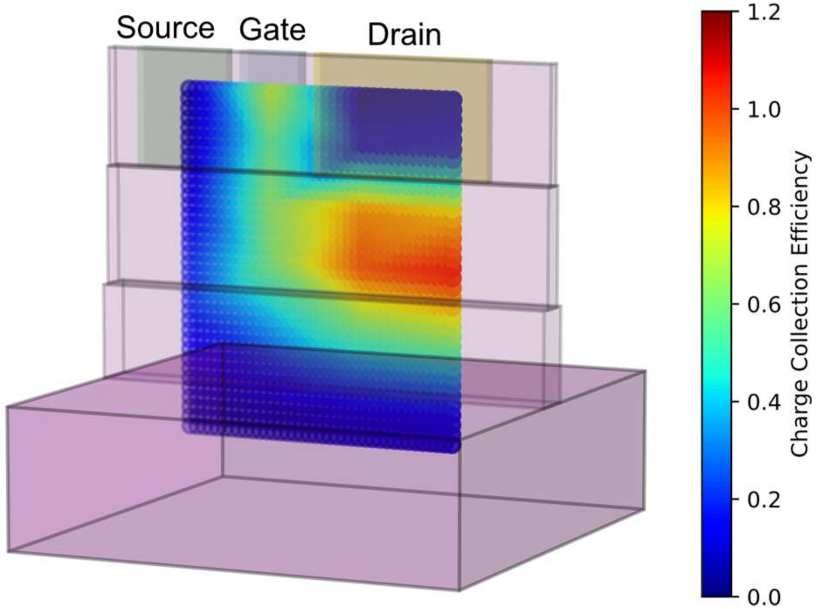


TCAD model

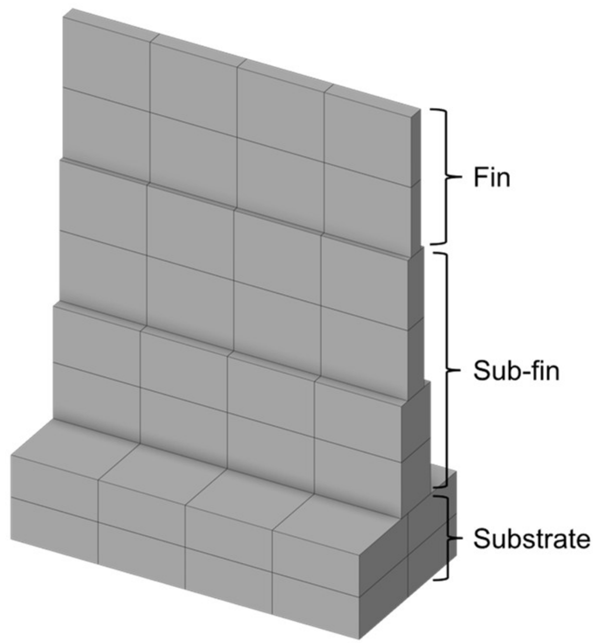
[9] N. Kotani, *SISPAD*, 1998.

Sensitive volumes allocation

- Charges deposited to individual SVs are input variables of the classifier.
- SVs are allocated to off-state NMOS.



Region (Ceff > 0.05) is considered for SV allocation



Allocated 40 SVs

ML setup

- LightGBM [10] is adopted.
- Split the dataset into an 8:2 ratio for training and testing, ensuring that the label distribution remains the same in both sets.

- Initial 10 samples for labeling is selected randomly.
- HP is updated every time # of labeled samples increases by 10.

[10] G. Ke, et al., *NIPS*, 2017.

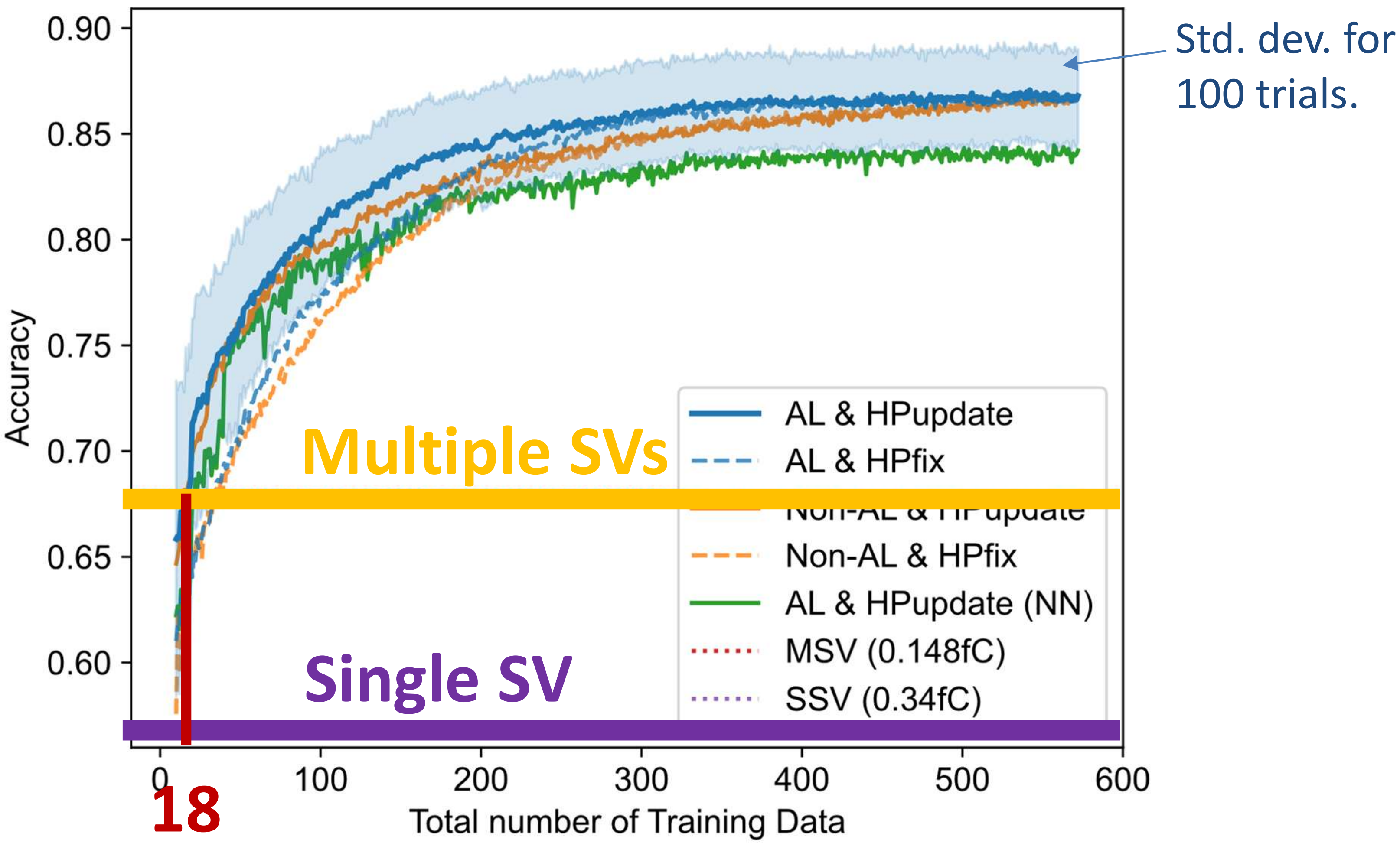
Procedure of hyperparameter update

1. Split the current training data into stratified 5-folds randomly.
2. Explore hyperparameters using Optuna [11]. For each fold, construct a discriminator using the other four folds and evaluate its accuracy.
3. Repeat Step 2 100 times and adopt the hyperparameters with the highest average accuracy across the 5-folds.

[11] T. Akiba, et al., *KDD*, 2019.

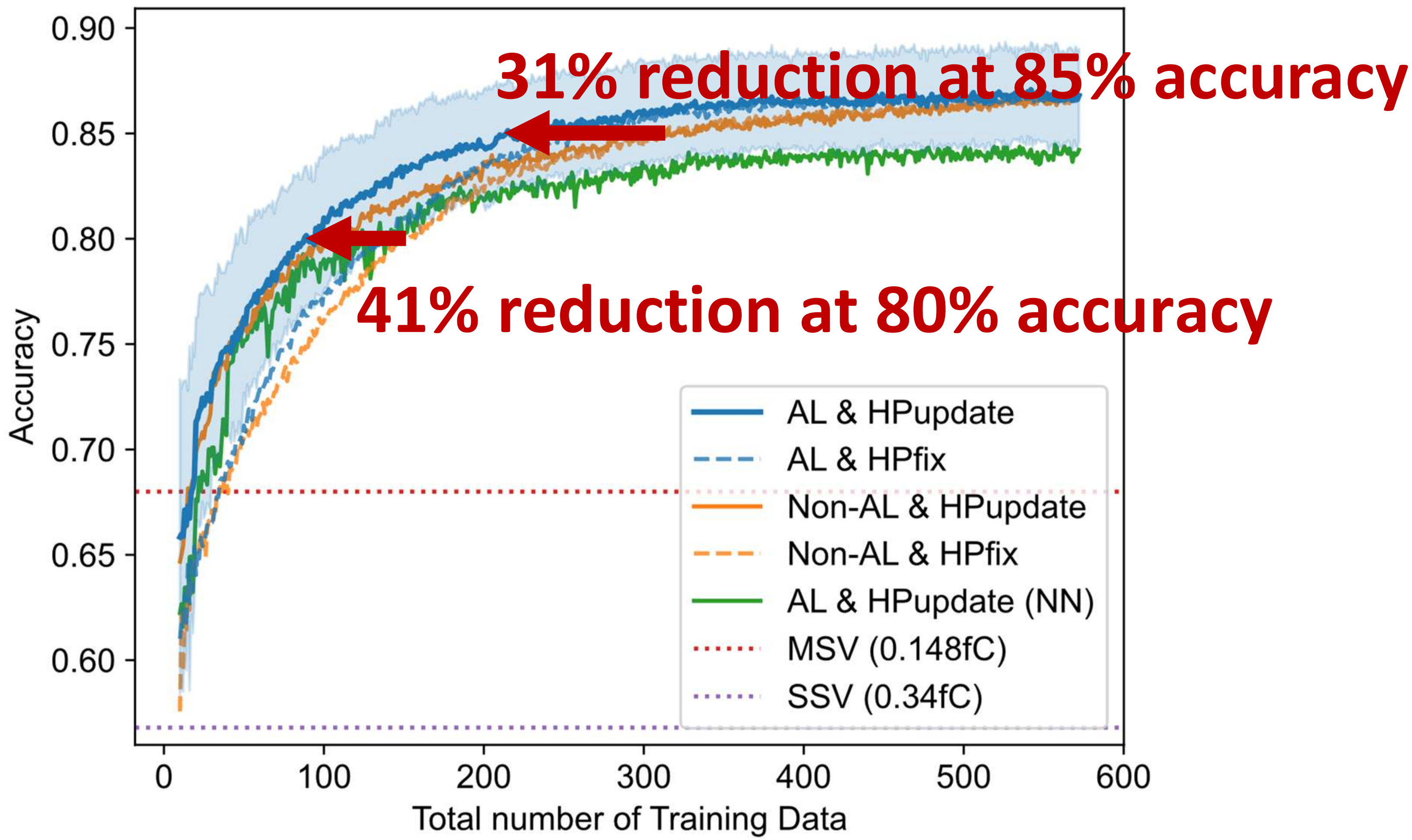
Accuracy vs. # of training data

ML-based classifiers outperform conventional SV methods.
Only 18 data to reach MSV accuracy.



Accuracy vs. # of training data

Compared with ML w/o AL and HP update,
the proposed method achieves



Runtime reduction

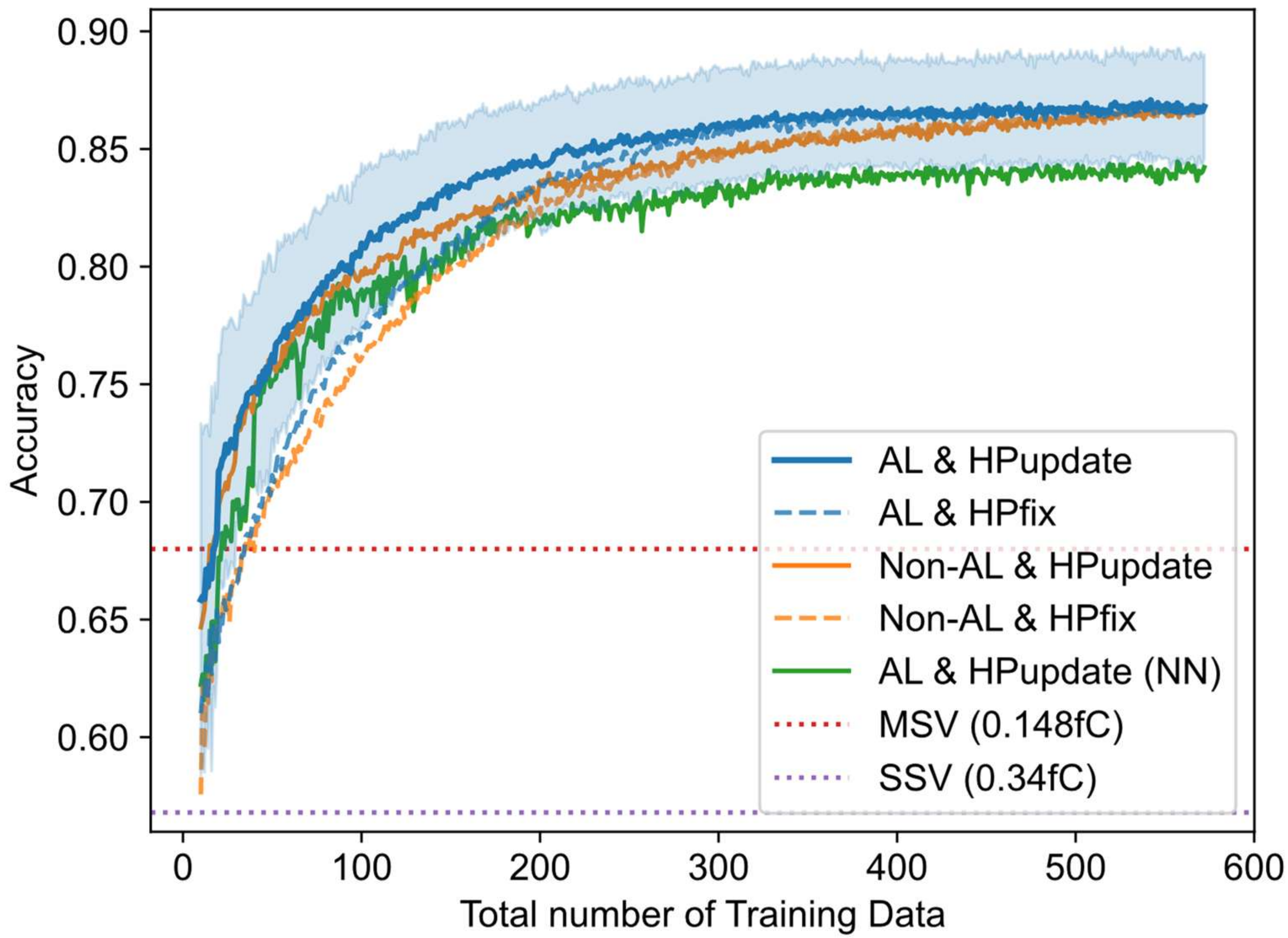
- 12 hours per one TCAD simulation in average
- 160 simulations (80% accuracy w/ conv. training) require 1,920 hours (80 days)

-> 30 to 40% reduction is significant.

- Compared with TCAD, training and hyperparameter tuning are negligible.

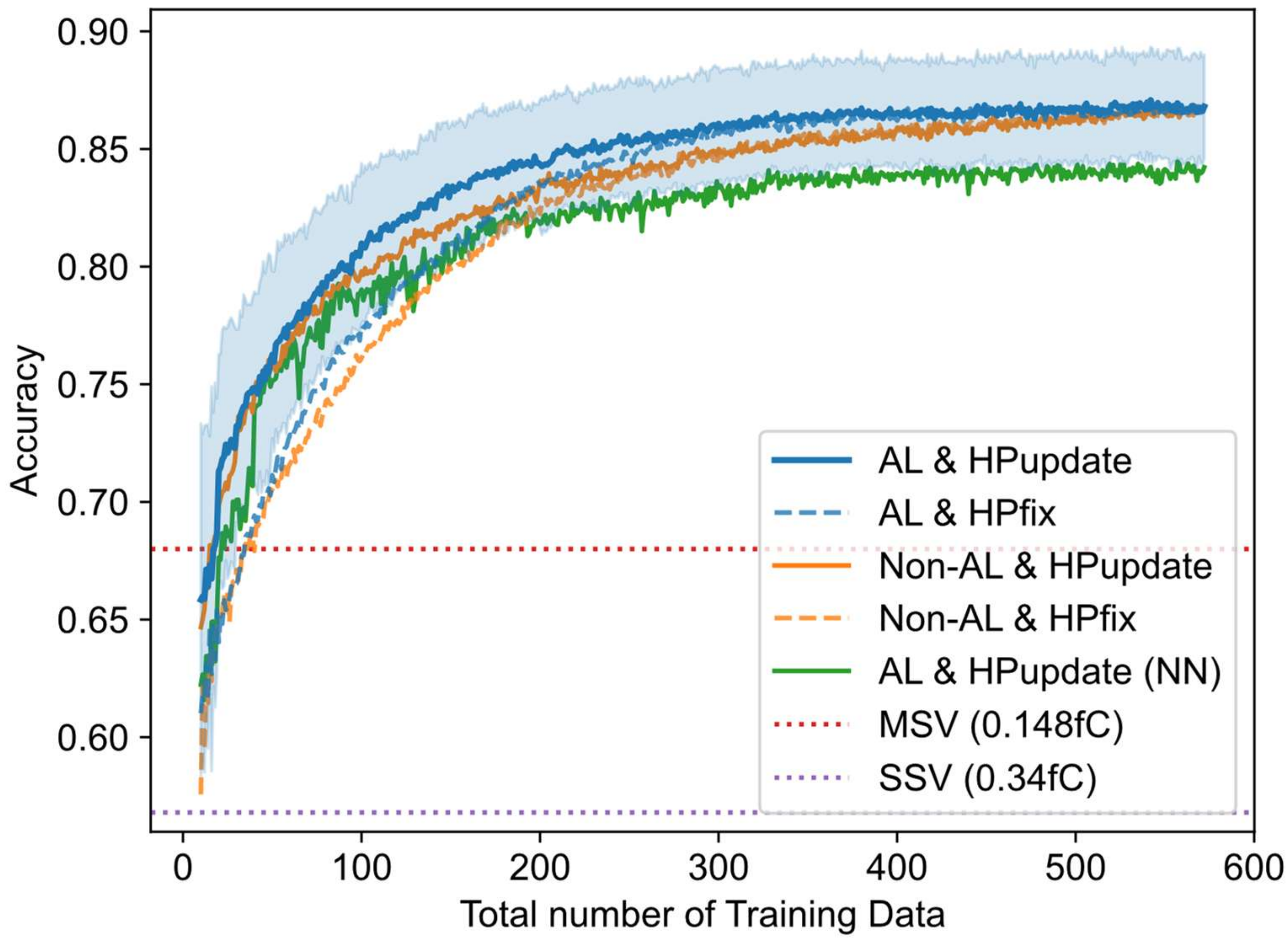
Accuracy vs. # of training data

AL vs non-AL

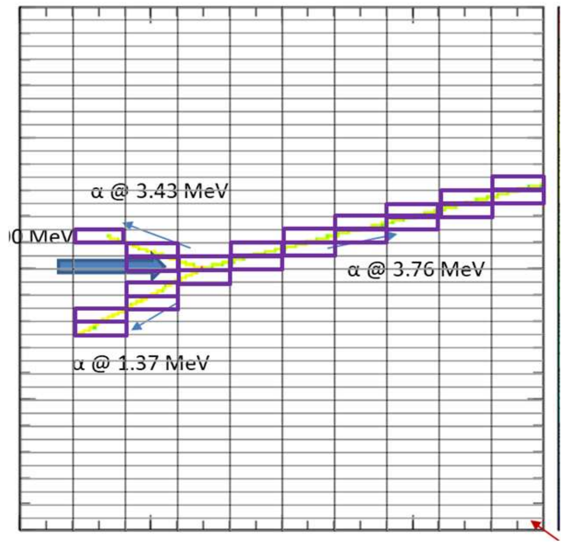


Accuracy vs. # of training data

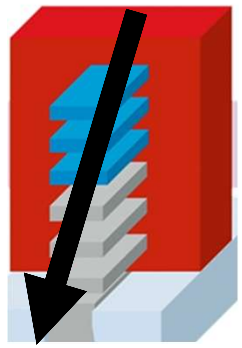
HPupdate (solid) vs HPfix (dashed)



Future directions



- Validate the proposed SER estimation using measurement data focusing on multiple cell upsets (MCUs).
- Extend the proposed method to account for particles striking multiple transistors within a cell since such events alter sensitivity to charge deposition [6]. GAA and CFET are likely to encounter these events.
- Accelerate TCAD simulations using ML [12]. TCAD could be directly integrated into the Monte Carlo flow.



[12] R. Novkin, et al., "ML-TCAD: Perspectives and Challenges on Accelerating Transistor Modeling using ML," *MLCAD*, 2023.

Conclusions

- Proposed an SEU discriminator construction method w/ active learning and periodic hyperparameter tuning
  - enhancing both the accuracy and training efficiency.
  - systematically constructing the discriminator w/o empirical determination of SV and critical charge.
- Applied to a 12-nm FinFET SRAM using LightGBM.
  - both active learning and hyperparameter updating effectively reduce the necessary training data
  - 41% and 31% reductions for 80% and 85% discrimination accuracy, respectively
- Discussed future directions