HapPyCal: A Fast Calorimeter Simulation Framework for the πM1 Beam Line

July 28, 2022

Zain Eris Kamal; Advised by Ronald Gilman and Wan Lin

Department of Physics and Astronomy, Rutgers University, Piscataway NJ

Aresty Research Center
for Undergraduates

HapPyCal: A Fast Calorimeter Simulation Framework for the πM1 Beam Line

RUTGERS Aresty Research Center for Undergraduates

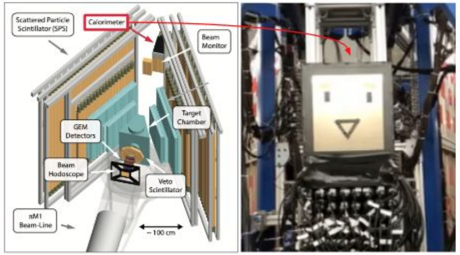
Zain Eris Kamal; Advised by Ronald Gilman and Wan Lin

Department of Physics and Astronomy, Rutgers University, Piscataway NJ

The Proton Radius Puzzle

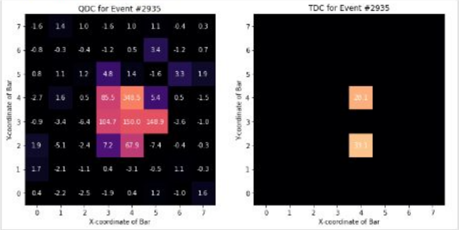
- In 2010, Pohl et al. measured the proton radius using *muonic* hydrogen spectroscopy and found it to be 5.6 σ smaller than the accepted value determined through *electronic* hydrogen. It is still unclear how much of this difference results from interesting physics vs. experimental error.
- The MUon Scattering Experiment (MUSE) performs simultaneous muon and electron scattering off of a liquid hydrogen target with the goal of measuring the proton radius with high-precision. This will also provide a test of lepton universality and the possibility of two-photon exchange by switching between beam polarities.

πM1 Calorimeter



The πM1 beam line at the Paul Scherrer Institute has an electromagnetic **calorimeter** (pictured above) which measures the ***energy and timing* of particles moving along the beam line**. When high-energy muons/pions/electrons enter the calorimeter, they initiate a "particle shower", a cascade of lower-energy secondary particles which deposit their energy into the calorimeter.

The calorimeter is an 8 by 8 array of independently read out crystals, which are monitored in 2 ways: QDC (Charge-to-Digital Converter) and TDC (Time-to-Digital Converter). We can visualize the data collected by the calorimeter for a single event in the colormaps below, with the "hotter" pixels representing particle impact sites.



Abstract

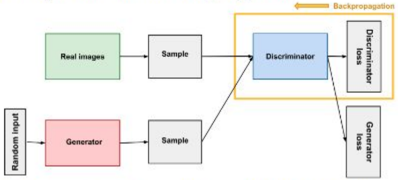
In high energy physics, detailed and accurate simulations of detectors (such as calorimeters) are the **single most expensive element of computational pipelines**. This is because most analyses require aggregated distributions of detector data as opposed to "realistic" data of individual events, creating heavy demand for fast simulation frameworks.

Traditionally ***GEANT4*** software is used to simulate detector responses, but this can take **minutes per individual scattering event** (of which millions are required) and is prone to systematic deviations from experimental data.

Thus we develop ***HapPyCal***, a fast detector simulation framework which can accurately reproduce calorimeter energy scans significantly faster than previous methods. We adapt a machine learning architecture called "Generative Adversarial Networks" and train it with data collected from experimental runs. Through analyses of energy scans from electron scattering events, we find that our model can **accurately reconstruct a larger distribution of data** with just **5%** of the original dataset.

Methodology

Generative Adversarial Networks (GANs) are a type of deep generative model characterized by two key components: a generator network and a discriminator network. The generator's goal is to create realistic data (in this case, calorimeter energy scans), while the discriminator's goal is to be able to tell if some given data is real or created by the generator.

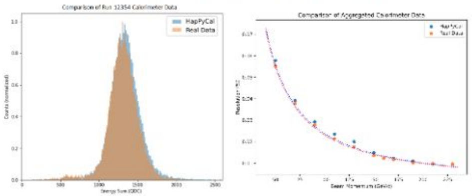


In the training process, **the generator repeatedly tries to "fool" the discriminator**. At first, the generator's outputs are recognized as fake every time, but by applying backpropagation algorithms, **the generator is able to "learn" from its mistakes** and gradually get better and better at producing "realistic" data. The discriminator is trained in tandem, creating a type of two-player minimax game.

On a more technical level, we use PyTorch to structure our generator as a **convolutional neural network** with three layers and a combination of ReLU and TanH activation functions. The discriminator is a standard **binary classification model** with a cross entropy loss function.

Results

After training our model on ~5% of the calorimeter data from each dataset over a target range of momenta, we **compare HapPyCal-generated data** (blue) with **real data** collected by the πM1 calorimeter (orange) below:



1. On the **left**, we focus on a *single* "run" (**fixed momentum:** 230 MeV/c) and show the distribution of total energy deposited by electron scattering events.
2. On the **right**, we look at *many* runs (**variable momentum:** 50-230 MeV/c) and show how the energy resolution (proportional to standard deviation) changes with momentum.

In both cases, the curve-fits for HapPyCal data and real data are incredibly close, suggesting that the generative modelling technique is **accurately "learning" the underlying distributions** of electron scattering data.

We note that in most of our analyses, HapPyCal's energy values tend to be greater than in real data. This can be attributed to the fact that GANs are usually susceptible to noisy outputs when the training data can't be normalized as in our case.

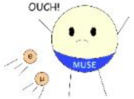
Conclusions / Further Work

The speed and accuracy of HapPyCal opens up some interesting opportunities for deeper analysis at MUSE. On the simulation side, analysis of GEANT4 models is now significantly faster. On the experimental side, we can also evaluate how detector responses vary over small periods to identity any drift in the data.

GANs are a powerful tool, but have many well known problems such noisy outputs and unstable/non convergent training. Future work with more cutting-edge techniques from machine learning literature such as Normalizing Flows and WGAN-GPs could offer even greater speed and accuracy.

Acknowledgements

This material is based upon work supported by Aresty and the National Science Foundation under NSF grant PHY-1913653. The MUSE experiment is supported by the Department of Energy, NSF, PSI, and the US-Israel Binational Science Foundation.



HapPyCal: A Fast Calorimeter Simulation Framework for the πM1 Beam Line

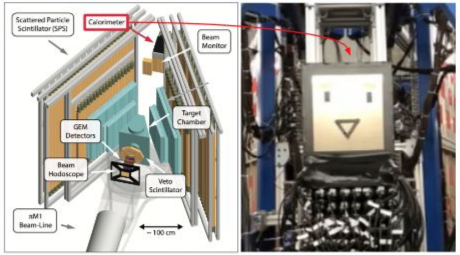
Zain Eris Kamal; Advised by Ronald Gilman and Wan Lin
Department of Physics and Astronomy, Rutgers University, Piscataway NJ

RUTGERS Aresty Research Center for Undergraduates

The Proton Radius Puzzle

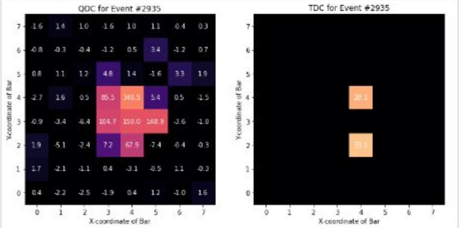
- In 2010, Pohl et al. measured the proton radius using *muonic* hydrogen spectroscopy and found it to be 5.6 σ smaller than the accepted value determined through *electronic* hydrogen. It is still unclear how much of this difference results from interesting physics vs. experimental error.
- The MUon Scattering Experiment (MUSE) performs simultaneous muon and electron scattering off of a liquid hydrogen target with the goal of measuring the proton radius with high-precision. This will also provide a test of lepton universality and the possibility of two-photon exchange by switching between beam polarities.

πM1 Calorimeter



The πM1 beam line at the Paul Scherrer Institute has an electromagnetic **calorimeter** (pictured above) which measures the ***energy and timing* of particles moving along the beam line**. When high-energy muons/pions/electrons enter the calorimeter, they initiate a "particle shower", a cascade of lower-energy secondary particles which deposit their energy into the calorimeter.

The calorimeter is an 8 by 8 array of independently read out crystals, which are monitored in 2 ways: QDC (Charge-to-Digital Converter) and TDC (Time-to-Digital Converter). We can visualize the data collected by the calorimeter for a single event in the colormaps below, with the "hotter" pixels representing particle impact sites.



Abstract

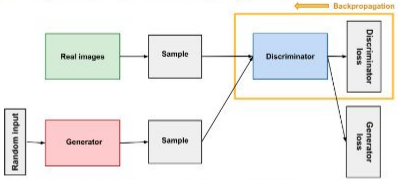
In high energy physics, detailed and accurate simulations of detectors (such as calorimeters) are the **single most expensive element of computational pipelines**. This is because most analyses require aggregated distributions of detector data as opposed to "realistic" data of individual events, creating heavy demand for fast simulation frameworks.

Traditionally ***GEANT4*** software is used to simulate detector responses, but this can take **minutes per individual scattering event** (of which millions are required) and is prone to systematic deviations from experimental data.

Thus we develop ***HapPyCal***, a fast detector simulation framework which can accurately reproduce calorimeter energy scans significantly faster than previous methods. We adapt a machine learning architecture called "Generative Adversarial Networks" and train it with data collected from experimental runs. Through analyses of energy scans from electron scattering events, we find that our model can **accurately reconstruct a larger distribution of data** with just **5%** of the original dataset.

Methodology

Generative Adversarial Networks (GANs) are a type of deep generative model characterized by two key components: a generator network and a discriminator network. The generator's goal is to create realistic data (in this case, calorimeter energy scans), while the discriminator's goal is to be able to tell if some given data is real or created by the generator.

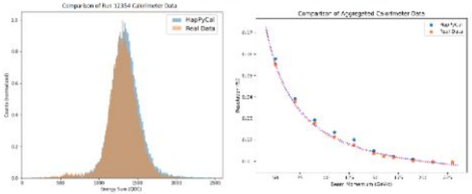


In the training process, **the generator repeatedly tries to "fool" the discriminator**. At first, the generator's outputs are recognized as fake every time, but by applying backpropagation algorithms, **the generator is able to "learn" from its mistakes** and gradually get better and better at producing "realistic" data. The discriminator is trained in tandem, creating a type of two-player minimax game.

On a more technical level, we use PyTorch to structure our generator as a **convolutional neural network** with three layers and a combination of ReLU and TanH activation functions. The discriminator is a standard **binary classification model** with a cross entropy loss function.

Results

After training our model on ~5% of the calorimeter data from each dataset over a target range of momenta, we **compare HapPyCal-generated data** (blue) with **real data** collected by the πM1 calorimeter (orange) below:



1. On the **left**, we focus on a *single* "run" (**fixed momentum:** 230 MeV/c) and show the distribution of total energy deposited by electron scattering events.
2. On the **right**, we look at *many* runs (**variable momentum:** 50-230 MeV/c) and show how the energy resolution (proportional to standard deviation) changes with momentum.

In both cases, the curve-fits for HapPyCal data and real data are incredibly close, suggesting that the generative modelling technique is **accurately "learning" the underlying distributions** of electron scattering data.

We note that in most of our analyses, HapPyCal's energy values tend to be greater than in real data. This can be attributed to the fact that GANs are usually susceptible to noisy outputs when the training data can't be normalized as in our case.

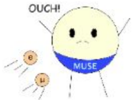
Conclusions / Further Work

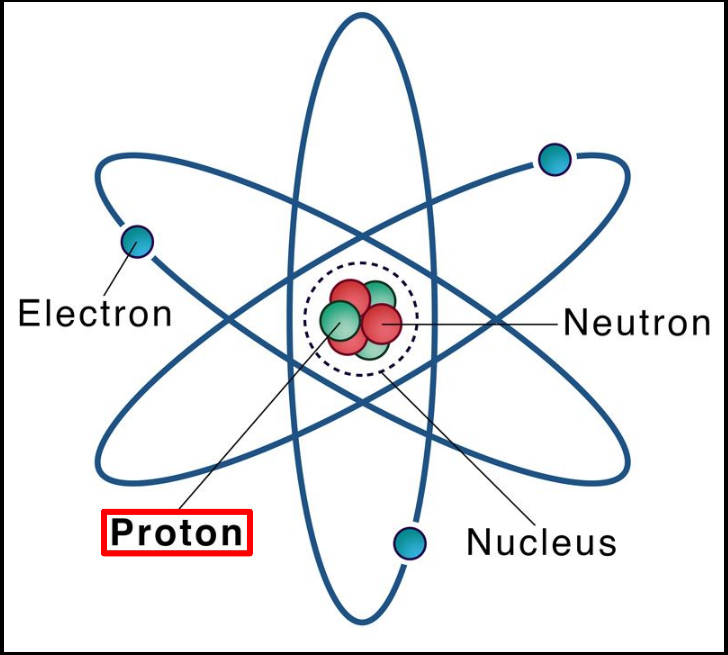
The speed and accuracy of HapPyCal opens up some interesting opportunities for deeper analysis at MUSE. On the simulation side, analysis of GEANT4 models is now significantly faster. On the experimental side, we can also evaluate how detector responses vary over small periods to identity any drift in the data.

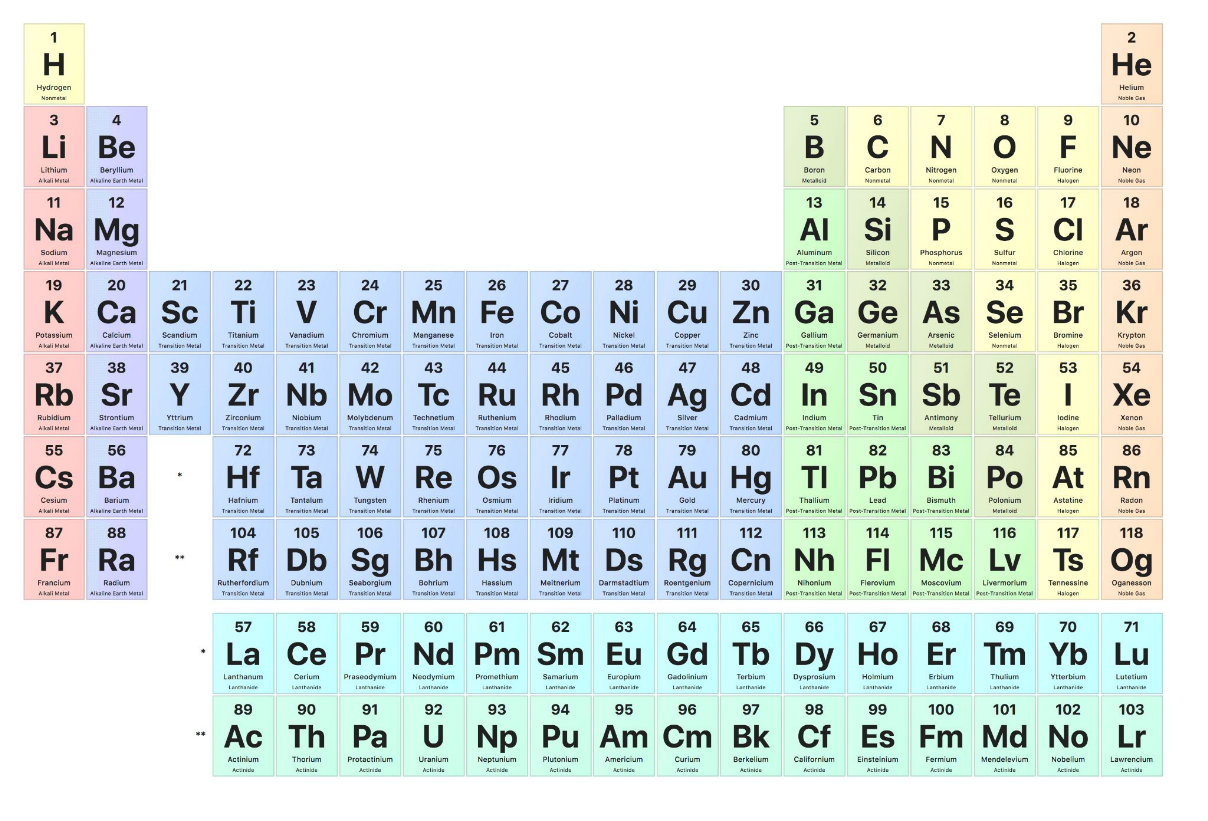
GANs are a powerful tool, but have many well known problems such noisy outputs and unstable/non convergent training. Future work with more cutting-edge techniques from machine learning literature such as Normalizing Flows and WGAN-GPs could offer even greater speed and accuracy.

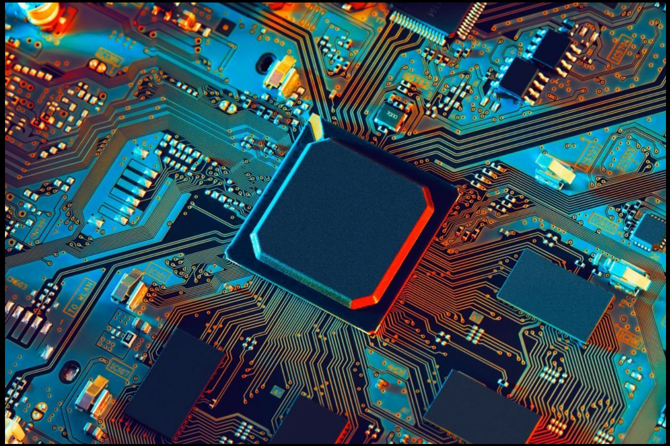
Acknowledgements

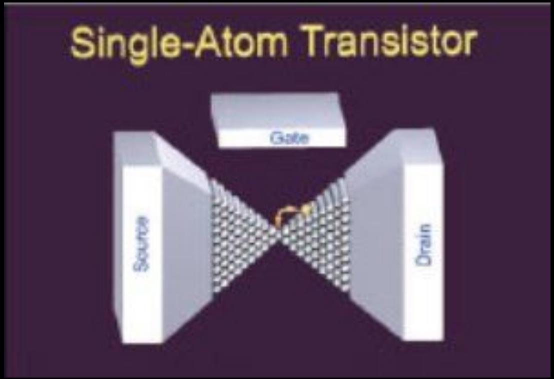
This material is based upon work supported by Aresty and the National Science Foundation under NSF grant PHY-1913653. The MUSE experiment is supported by the Department of Energy, NSF, PSI, and the US-Israel Binational Science Foundation.

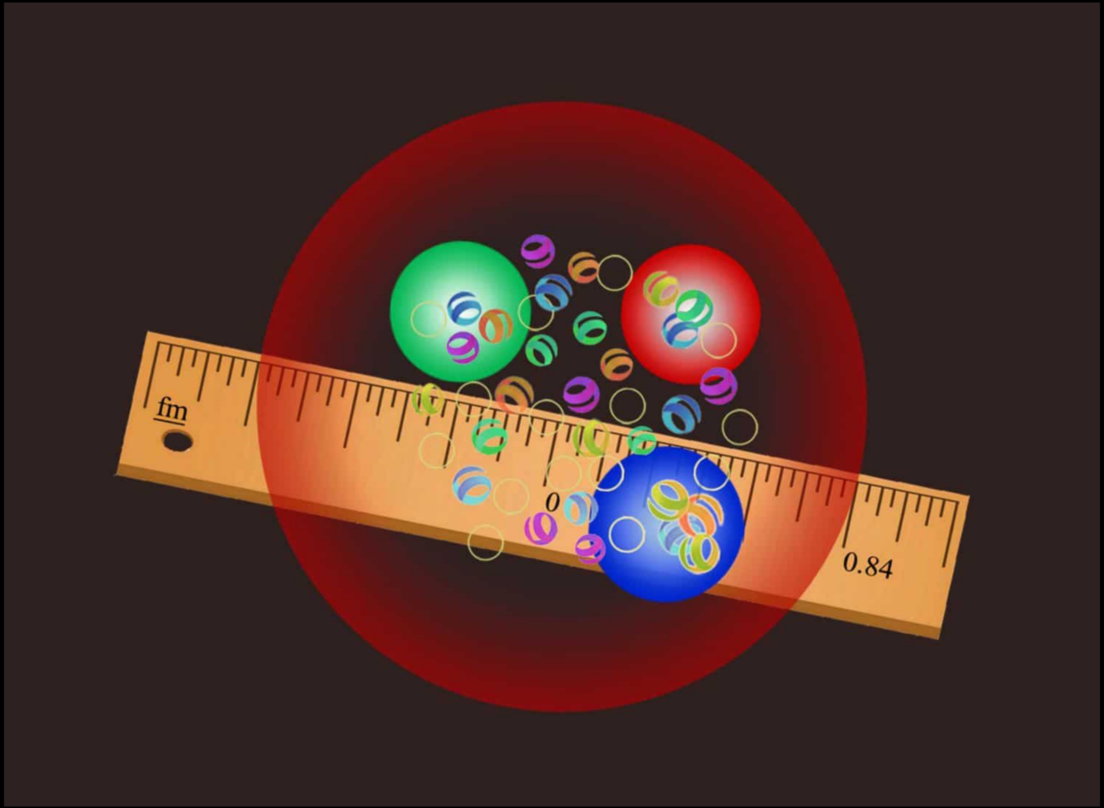


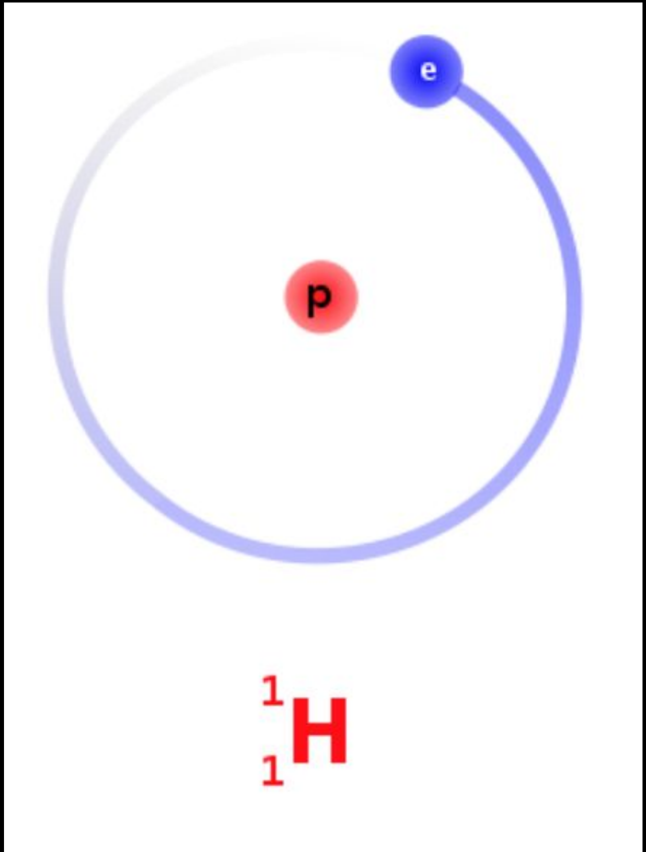

Electron
Neutron
Proton
Nucleus


1 H Hydrogen Nonmetal
2 He Helium Noble Gas
3 Li Lithium Alkali Metal
4 Be Beryllium Alkaline Earth Metal
5 B Boron Metalloid
6 C Carbon Nonmetal
7 N Nitrogen Nonmetal
8 O Oxygen Nonmetal
9 F Fluorine Halogen
10 Ne Neon Noble Gas
11 Na Sodium Alkali Metal
12 Mg Magnesium Alkaline Earth Metal
13 Al Aluminum Post-Transition Metal
14 Si Silicon Metalloid
15 P Phosphorus Nonmetal
16 S Sulfur Nonmetal
17 Cl Chlorine Halogen
18 Ar Argon Noble Gas
19 K Potassium Alkali Metal
20 Ca Calcium Alkaline Earth Metal
21 Sc Scandium Transition Metal
22 Ti Titanium Transition Metal
23 V Vanadium Transition Metal
24 Cr Chromium Transition Metal
25 Mn Manganese Transition Metal
26 Fe Iron Transition Metal
27 Co Cobalt Transition Metal
28 Ni Nickel Transition Metal
29 Cu Copper Transition Metal
30 Zn Zinc Transition Metal
31 Ga Gallium Post-Transition Metal
32 Ge Germanium Metalloid
33 As Arsenic Metalloid
34 Se Selenium Nonmetal
35 Br Bromine Halogen
36 Kr Krypton Noble Gas
37 Rb Rubidium Alkali Metal
38 Sr Strontium Alkaline Earth Metal
39 Y Yttrium Transition Metal
40 Zr Zirconium Transition Metal
41 Nb Niobium Transition Metal
42 Mo Molybdenum Transition Metal
43 Tc Technetium Transition Metal
44 Ru Ruthenium Transition Metal
45 Rh Rhodium Transition Metal
46 Pd Palladium Transition Metal
47 Ag Silver Transition Metal
48 Cd Cadmium Transition Metal
49 In Indium Post-Transition Metal
50 Sn Tin Post-Transition Metal
51 Sb Antimony Metalloid
52 Te Tellurium Metalloid
53 I Iodine Halogen
54 Xe Xenon Noble Gas
55 Cs Cesium Alkali Metal
56 Ba Barium Alkaline Earth Metal
*
72 Hf Hafnium Transition Metal
73 Ta Tantalum Transition Metal
74 W Tungsten Transition Metal
75 Re Rhenium Transition Metal
76 Os Osmium Transition Metal
77 Ir Iridium Transition Metal
78 Pt Platinum Transition Metal
79 Au Gold Transition Metal
80 Hg Mercury Transition Metal
81 Tl Thallium Post-Transition Metal
82 Pb Lead Post-Transition Metal
83 Bi Bismuth Post-Transition Metal
84 Po Polonium Metalloid
85 At Astatine Halogen
86 Rn Radon Noble Gas
87 Fr Francium Alkali Metal
88 Ra Radium Alkaline Earth Metal
**
104 Rf Rutherfordium Transition Metal
105 Db Dubnium Transition Metal
106 Sg Seaborgium Transition Metal
107 Bh Bohrium Transition Metal
108 Hs Hassium Transition Metal
109 Mt Meitnerium Transition Metal
110 Ds Darmstadtium Transition Metal
111 Rg Roentgenium Transition Metal
112 Cn Copernicium Transition Metal
113 Nh Nihonium Post-Transition Metal
114 Fl Flerovium Post-Transition Metal
115 Mc Moscovium Post-Transition Metal
116 Lv Livermorium Post-Transition Metal
117 Ts Tennessine Halogen
118 Og Oganesson Noble Gas
*
57 La Lanthanum Lanthanide
58 Ce Cerium Lanthanide
59 Pr Praseodymium Lanthanide
60 Nd Neodymium Lanthanide
61 Pm Promethium Lanthanide
62 Sm Samarium Lanthanide
63 Eu Europium Lanthanide
64 Gd Gadolinium Lanthanide
65 Tb Terbium Lanthanide
66 Dy Dysprosium Lanthanide
67 Ho Holmium Lanthanide
68 Er Erbium Lanthanide
69 Tm Thulium Lanthanide
70 Yb Ytterbium Lanthanide
71 Lu Lutetium Lanthanide
**
89 Ac Actinium Actinide
90 Th Thorium Actinide
91 Pa Protactinium Actinide
92 U Uranium Actinide
93 Np Neptunium Actinide
94 Pu Plutonium Actinide
95 Am Americium Actinide
96 Cm Curium Actinide
97 Bk Berkelium Actinide
98 Cf Californium Actinide
99 Es Einsteinium Actinide
100 Fm Fermium Actinide
101 Md Mendelevium Actinide
102 No Nobelium Actinide
103 Lr Lawrencium Actinide

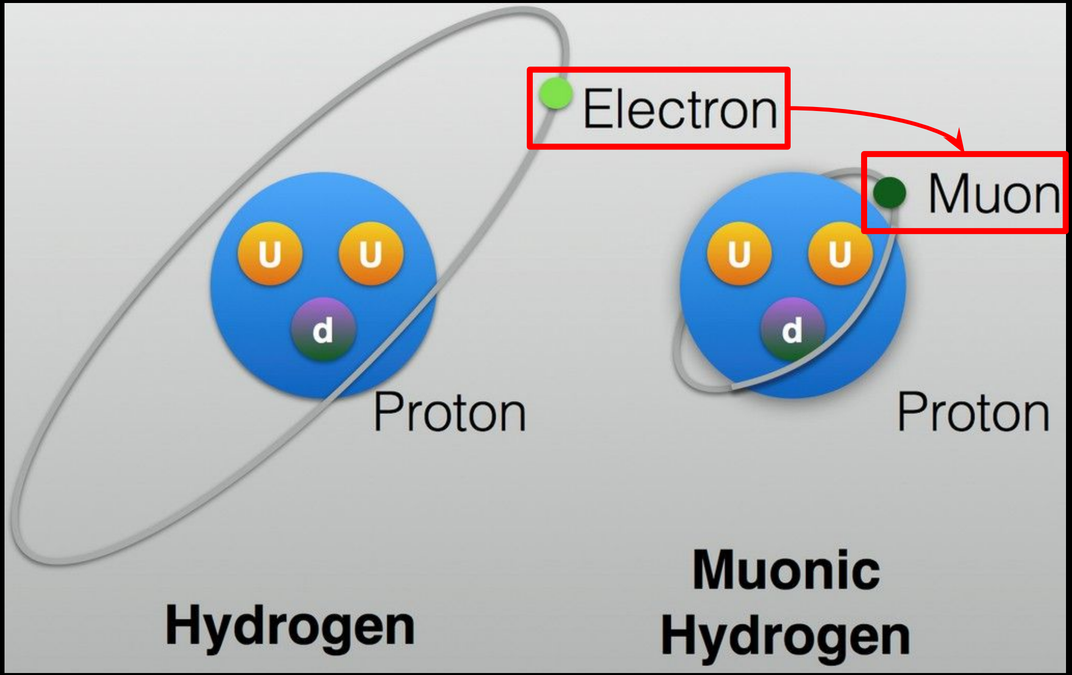


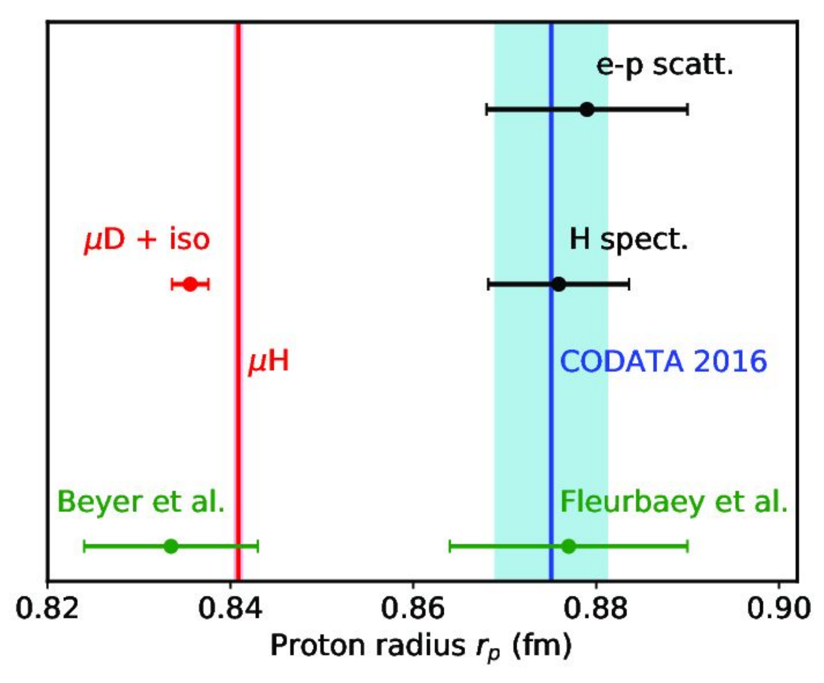

Single-Atom Transistor
Gate
Source
Drain


fm
0
0.84



Early proton radius = 0.87 fm


Electron
Muon
U
U
d
Proton
U
U
d
Proton
Hydrogen
Muonic
Hydrogen



(old measurements on right, new muonic hydrogen on left)

The Proton Radius Puzzle


8 July 2010 | www.nature.com/nature | £10

THE INTERNATIONAL WEEKLY JOURNAL OF SCIENCE

nature

OIL SPILLS
There's more to come

PLAGIARISM
It's worse than you think

CHIMPANZEES
The battle for survival

0.84184 fm

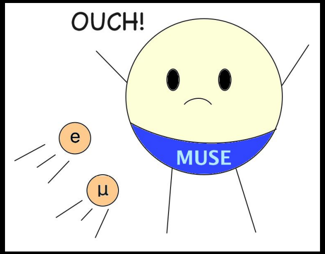
SHRINKING THE PROTON

New value from exotic atom trims radius by four per cent

NATUREJOBS
Researchers for hire

MUSE

"MUon Scattering Experiment"



HapPyCal: A Fast Calorimeter Simulation Framework for the πM1 Beam Line

Rutgers Aresty Research Center for Undergraduates

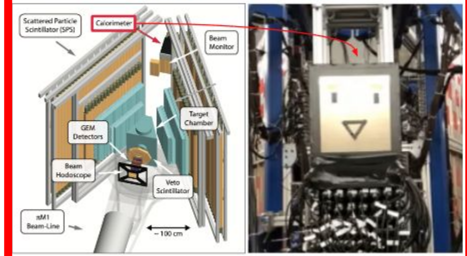
Zain Eris Kamal; Advised by Ronald Gilman and Wan Lin

Department of Physics and Astronomy, Rutgers University, Piscataway NJ

The Proton Radius Puzzle

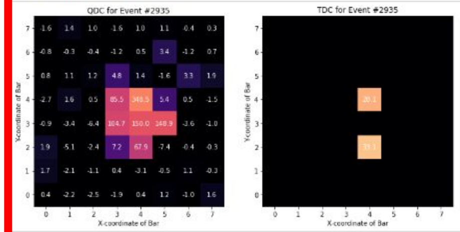
- In 2010, Pohl et al. measured the proton radius using *muonic* hydrogen spectroscopy and found it to be **5.6 σ smaller** than the accepted value determined through *electronic* hydrogen. It is still unclear how much of this difference results from interesting physics vs. experimental error.
- The **MUon Scattering Experiment (MUSE)** performs simultaneous muon and electron scattering off of a liquid hydrogen target with the goal of measuring the proton radius with high-precision. This will also provide a test of lepton universality and the possibility of two-photon exchange by switching between beam polarities.

πM1 Calorimeter



The πM1 beam line at the Paul Scherrer Institute has an electromagnetic **calorimeter** (pictured above) which measures the ***energy and timing* of particles moving along the beam line**. When high-energy muons/pions/electrons enter the calorimeter, they initiate a "particle shower", a cascade of lower-energy secondary particles which deposit their energy into the calorimeter.

The calorimeter is an 8 by 8 array of independently read out crystals, which are monitored in 2 ways: QDC (Charge-to-Digital Converter) and TDC (Time-to-Digital Converter). We can visualize the data collected by the calorimeter for a single event in the colormaps below, with the "hotter" pixels representing particle impact sites.



Abstract

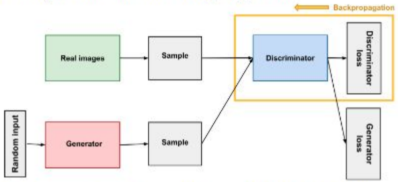
In high energy physics, detailed and accurate simulations of detectors (such as calorimeters) are the **single most expensive element of computational pipelines**. This is because most analyses require aggregated distributions of detector data as opposed to "realistic" data of individual events, creating heavy demand for fast simulation frameworks.

Traditionally ***GEANT4*** software is used to simulate detector responses, but this can take **minutes per individual scattering event** (of which millions are required) and is prone to systematic deviations from experimental data.

Thus we develop ***HapPyCal***, a fast detector simulation framework which can accurately reproduce calorimeter energy scans significantly faster than previous methods. We adapt a machine learning architecture called "Generative Adversarial Networks" and train it with data collected from experimental runs. Through analyses of energy scans from electron scattering events, we find that our model can **accurately reconstruct a larger distribution of data** with just **5%** of the original dataset.

Methodology

Generative Adversarial Networks (GANs) are a type of deep generative model characterized by two key components: a generator network and a discriminator network. The generator's goal is to create realistic data (in this case, calorimeter energy scans), while the discriminator's goal is to be able to tell if some given data is real or created by the generator.

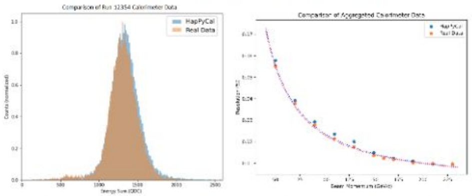


In the training process, **the generator repeatedly tries to "fool" the discriminator**. At first, the generator's outputs are recognized as fake every time, but by applying backpropagation algorithms, **the generator is able to "learn" from its mistakes** and gradually get better and better at producing "realistic" data. The discriminator is trained in tandem, creating a type of two-player minimax game.

On a more technical level, we use PyTorch to structure our generator as a **convolutional neural network** with three layers and a combination of ReLU and TanH activation functions. The discriminator is a standard **binary classification model** with a cross entropy loss function.

Results

After training our model on ~5% of the calorimeter data from each dataset over a target range of momenta, we **compare HapPyCal-generated data** (blue) with **real data** collected by the πM1 calorimeter (orange) below:



1. On the **left**, we focus on a *single* "run" (**fixed momentum:** 230 MeV/c) and show the distribution of total energy deposited by electron scattering events.
2. On the **right**, we look at *many* runs (**variable momentum:** 50-230 MeV/c) and show how the energy resolution (proportional to standard deviation) changes with momentum.

In both cases, the curve-fits for HapPyCal data and real data are incredibly close, suggesting that the generative modelling technique is **accurately "learning" the underlying distributions** of electron scattering data.

We note that in most of our analyses, HapPyCal's energy values tend to be greater than in real data. This can be attributed to the fact that GANs are usually susceptible to noisy outputs when the training data can't be normalized as in our case.

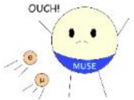
Conclusions / Further Work

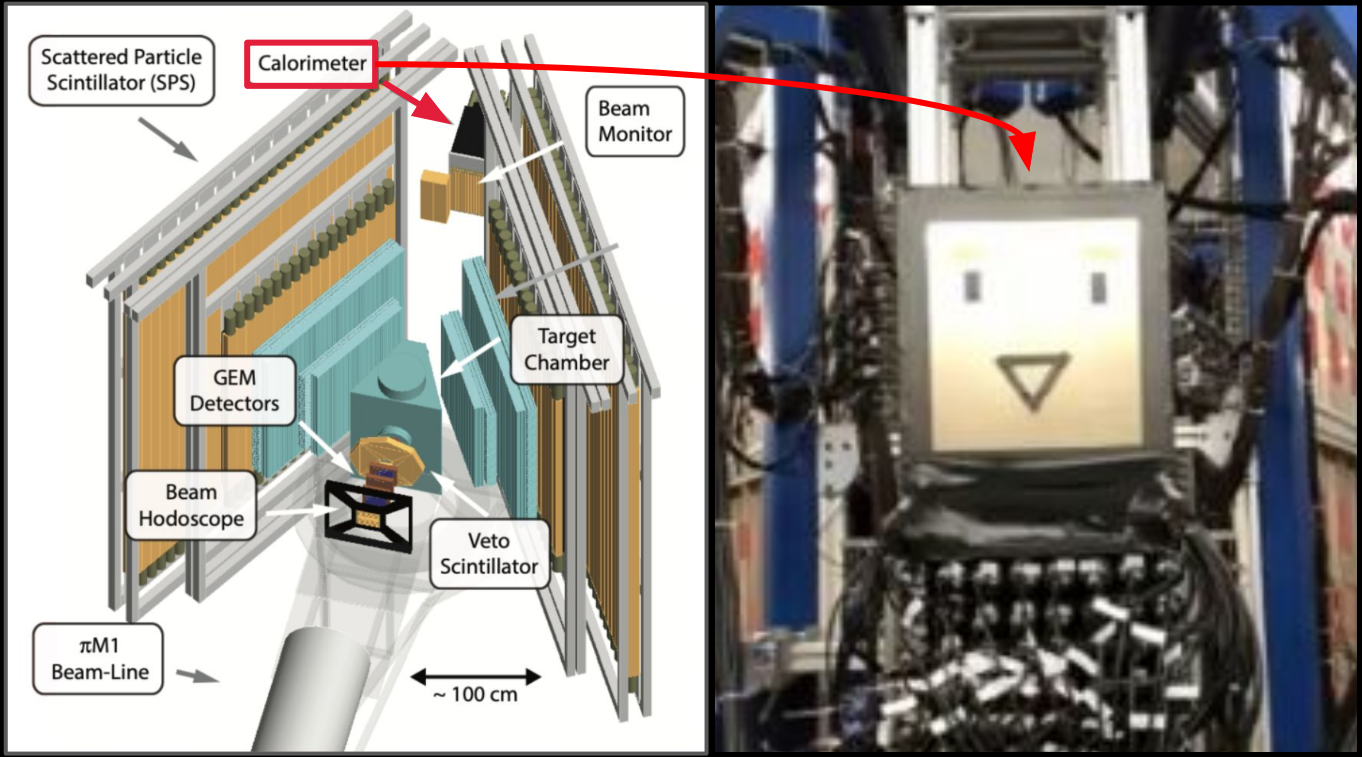
The speed and accuracy of HapPyCal opens up some interesting opportunities for deeper analysis at MUSE. On the simulation side, analysis of GEANT4 models is now significantly faster. On the experimental side, we can also evaluate how detector responses vary over small periods to identity any drift in the data.

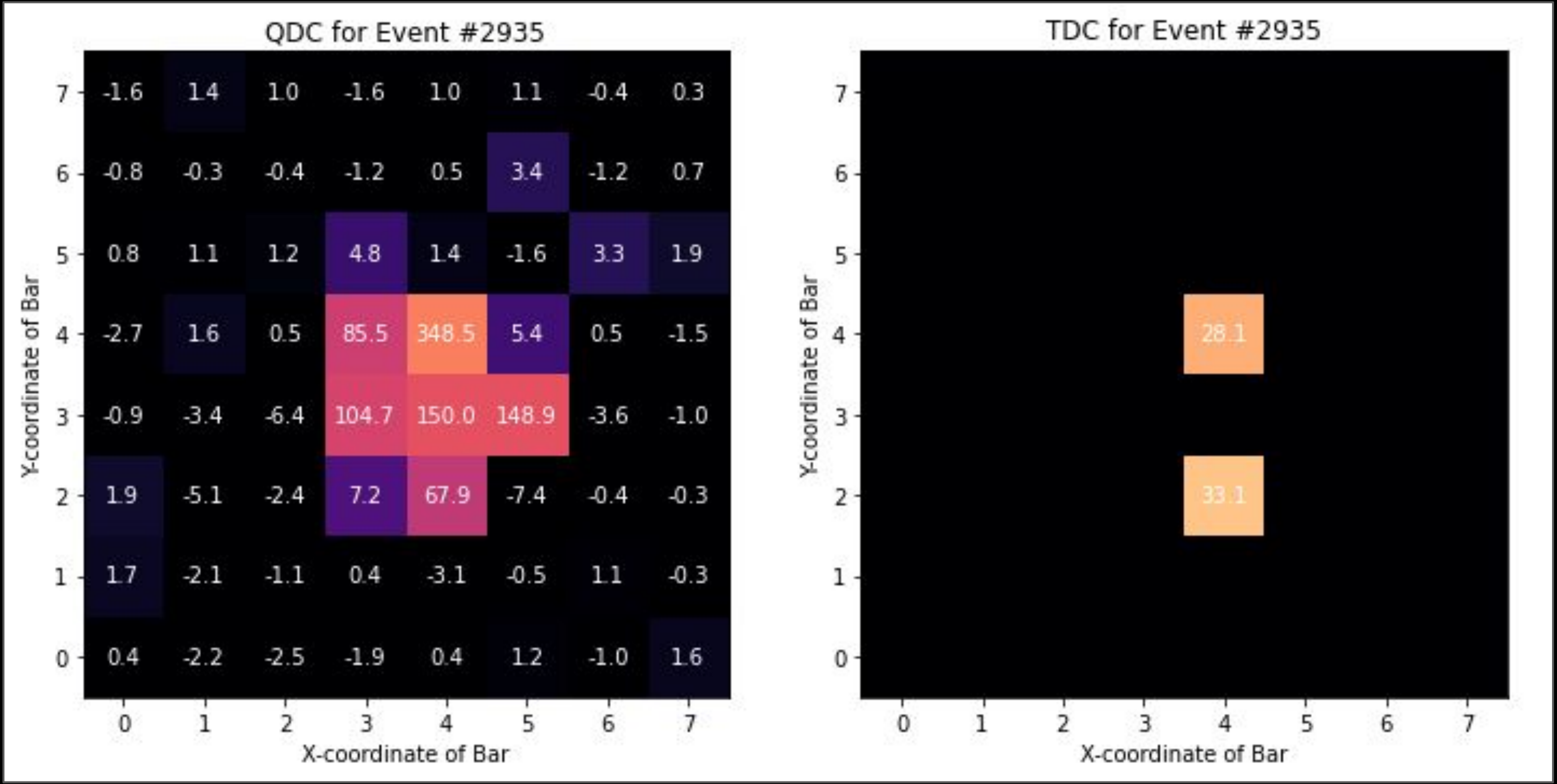
GANs are a powerful tool, but have many well known problems such noisy outputs and unstable/non convergent training. Future work with more cutting-edge techniques from machine learning literature such as Normalizing Flows and WGAN-GPs could offer even greater speed and accuracy.

Acknowledgements

This material is based upon work supported by Aresty and the National Science Foundation under NSF grant PHY-1913653. The MUSE experiment is supported by the Department of Energy, NSF, PSI, and the US-Israel Binational Science Foundation.




Scattered Particle Scintillator (SPS)
Calorimeter
Beam Monitor
Target Chamber
GEM Detectors
Beam Hodoscope
Veto Scintillator
πM1 Beam-Line
~ 100 cm


QDC for Event #2935
TDC for Event #2935
Y-coordinate of Bar
X-coordinate of Bar
-1.6 1.4 1.0 -1.6 1.0 1.1 -0.4 0.3
-0.8 -0.3 -0.4 -1.2 0.5 3.4 -1.2 0.7
0.8 1.1 1.2 4.8 1.4 -1.6 3.3 1.9
-2.7 1.6 0.5 85.5 348.5 5.4 0.5 -1.5
-0.9 -3.4 -6.4 104.7 150.0 148.9 -3.6 -1.0
1.9 -5.1 -2.4 7.2 67.9 -7.4 -0.4 -0.3
1.7 -2.1 -1.1 0.4 -3.1 -0.5 1.1 -0.3
0.4 -2.2 -2.5 -1.9 0.4 1.2 -1.0 1.6
28.1
33.1

HapPyCal: A Fast Calorimeter Simulation Framework for the πM1 Beam Line

Rutgers Aresty Research Center for Undergraduates

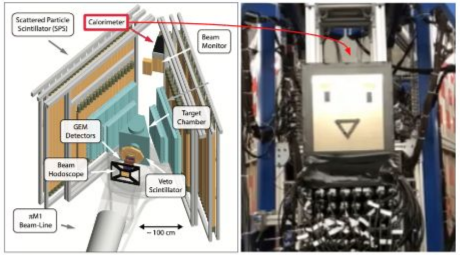
Zain Eris Kamal; Advised by Ronald Gilman and Wan Lin

Department of Physics and Astronomy, Rutgers University, Piscataway NJ

The Proton Radius Puzzle

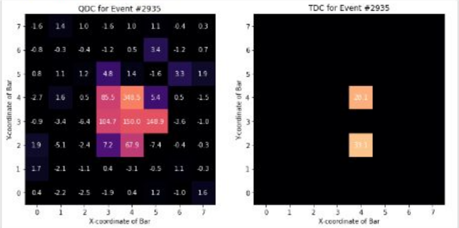
- In 2010, Pohl et al. measured the proton radius using *muonic* hydrogen spectroscopy and found it to be 5.6 σ smaller than the accepted value determined through *electronic* hydrogen. It is still unclear how much of this difference results from interesting physics vs. experimental error.
- The MUon Scattering Experiment (MUSE) performs simultaneous muon and electron scattering off of a liquid hydrogen target with the goal of measuring the proton radius with high-precision. This will also provide a test of lepton universality and the possibility of two-photon exchange by switching between beam polarities.

πM1 Calorimeter



The πM1 beam line at the Paul Scherrer Institute has an electromagnetic **calorimeter** (pictured above) which measures the ***energy and timing* of particles moving along the beam line**. When high-energy muons/pions/electrons enter the calorimeter, they initiate a "particle shower", a cascade of lower-energy secondary particles which deposit their energy into the calorimeter.

The calorimeter is an 8 by 8 array of independently read out crystals, which are monitored in 2 ways: QDC (Charge-to-Digital Converter) and TDC (Time-to-Digital Converter). We can visualize the data collected by the calorimeter for a single event in the colormaps below, with the "hotter" pixels representing particle impact sites.



Abstract

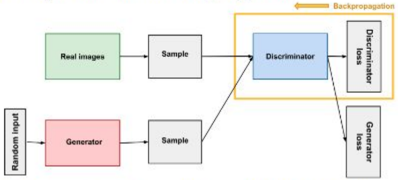
In high energy physics, detailed and accurate simulations of detectors (such as calorimeters) are the **single most expensive element of computational pipelines**. This is because most analyses require aggregated distributions of detector data as opposed to "realistic" data of individual events, creating heavy demand for fast simulation frameworks.

Traditionally ***GEANT4*** software is used to simulate detector responses, but this can take **minutes per individual scattering event** (of which millions are required) and is prone to systematic deviations from experimental data.

Thus we develop ***HapPyCal***, a fast detector simulation framework which can accurately reproduce calorimeter energy scans significantly faster than previous methods. We adapt a machine learning architecture called "Generative Adversarial Networks" and train it with data collected from experimental runs. Through analyses of energy scans from electron scattering events, we find that our model can **accurately reconstruct a larger distribution of data** with just **5%** of the original dataset.

Methodology

Generative Adversarial Networks (GANs) are a type of deep generative model characterized by two key components: a generator network and a discriminator network. The generator's goal is to create realistic data (in this case, calorimeter energy scans), while the discriminator's goal is to be able to tell if some given data is real or created by the generator.

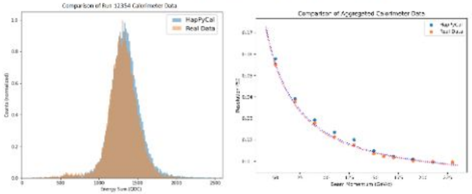


In the training process, **the generator repeatedly tries to "fool" the discriminator**. At first, the generator's outputs are recognized as fake every time, but by applying backpropagation algorithms, **the generator is able to "learn" from its mistakes** and gradually get better and better at producing "realistic" data. The discriminator is trained in tandem, creating a type of two-player minimax game.

On a more technical level, we use PyTorch to structure our generator as a **convolutional neural network** with three layers and a combination of ReLU and TanH activation functions. The discriminator is a standard **binary classification model** with a cross entropy loss function.

Results

After training our model on ~5% of the calorimeter data from each dataset over a target range of momenta, we **compare HapPyCal-generated data** (blue) with **real data** collected by the πM1 calorimeter (orange) below:



1. On the **left**, we focus on a *single* "run" (**fixed momentum:** 230 MeV/c) and show the distribution of total energy deposited by electron scattering events.
2. On the **right**, we look at *many* runs (**variable momentum:** 50-230 MeV/c) and show how the energy resolution (proportional to standard deviation) changes with momentum.

In both cases, the curve-fits for HapPyCal data and real data are incredibly close, suggesting that the generative modelling technique is **accurately "learning" the underlying distributions** of electron scattering data.

We note that in most of our analyses, HapPyCal's energy values tend to be greater than in real data. This can be attributed to the fact that GANs are usually susceptible to noisy outputs when the training data can't be normalized as in our case.

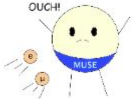
Conclusions / Further Work

The speed and accuracy of HapPyCal opens up some interesting opportunities for deeper analysis at MUSE. On the simulation side, analysis of GEANT4 models is now significantly faster. On the experimental side, we can also evaluate how detector responses vary over small periods to identity any drift in the data.

GANs are a powerful tool, but have many well known problems such noisy outputs and unstable/non convergent training. Future work with more cutting-edge techniques from machine learning literature such as Normalizing Flows and WGAN-GPs could offer even greater speed and accuracy.

Acknowledgements

This material is based upon work supported by Aresty and the National Science Foundation under NSF grant PHY-1913653. The MUSE experiment is supported by the Department of Energy, NSF, PSI, and the US-Israel Binational Science Foundation.



RUTGERS
Aresty Research Center for Undergraduates

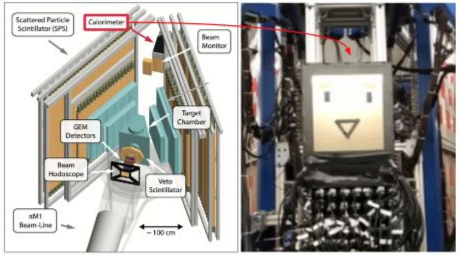
HapPyCal: A Fast Calorimeter Simulation Framework for the πM1 Beam Line

Zain Eris Kamal; Advised by Ronald Gilman and Wan Lin
Department of Physics and Astronomy, Rutgers University, Piscataway NJ

The Proton Radius Puzzle

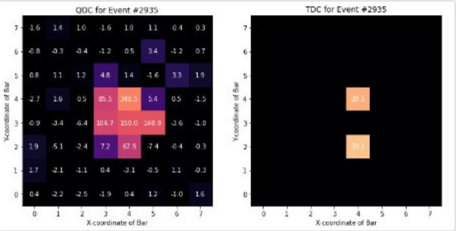
- In 2010, Pohl et al. measured the proton radius using *muonic* hydrogen spectroscopy and found it to be **5.6 σ smaller** than the accepted value determined through *electronic* hydrogen. It is still unclear how much of this difference results from interesting physics vs. experimental error.
- The **MUon Scattering Experiment (MUSE)** performs simultaneous muon and electron scattering off of a liquid hydrogen target with the goal of measuring the proton radius with high-precision. This will also provide a test of lepton universality and the possibility of two-photon exchange by switching between beam polarities.

πM1 Calorimeter



The πM1 beam line at the Paul Scherrer Institute has an electromagnetic **calorimeter** (pictured above) which measures the ***energy and timing* of particles moving along the beam line**. When high-energy muons/pions/electrons enter the calorimeter, they initiate a "particle shower", a cascade of lower-energy secondary particles which deposit their energy into the calorimeter.

The calorimeter is an 8 by 8 array of independently read out crystals, which are monitored in 2 ways: QDC (Charge-to-Digital Converter) and TDC (Time-to-Digital Converter). We can visualize the data collected by the calorimeter for a single event in the colormaps below, with the "hotter" pixels representing particle impact sites.



Abstract

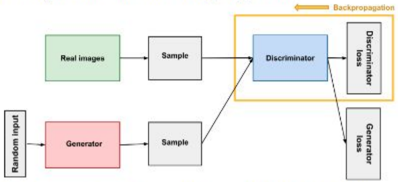
In high energy physics, detailed and accurate simulations of detectors (such as calorimeters) are the **single most expensive element of computational pipelines**. This is because most analyses require aggregated distributions of detector data as opposed to "realistic" data of individual events, creating heavy demand for fast simulation frameworks.

Traditionally ***GEANT4*** software is used to simulate detector responses, but this can take **minutes per individual scattering event** (of which millions are required) and is prone to systematic deviations from experimental data.

Thus we develop ***HapPyCal***, a fast detector simulation framework which can accurately reproduce calorimeter energy scans significantly faster than previous methods. We adapt a machine learning architecture called "Generative Adversarial Networks" and train it with data collected from experimental runs. Through analyses of energy scans from electron scattering events, we find that our model can **accurately reconstruct a larger distribution of data** with just **5%** of the original dataset.

Methodology

Generative Adversarial Networks (GANs) are a type of deep generative model characterized by two key components: a generator network and a discriminator network. The generator's goal is to create realistic data (in this case, calorimeter energy scans), while the discriminator's goal is to be able to tell if some given data is real or created by the generator.

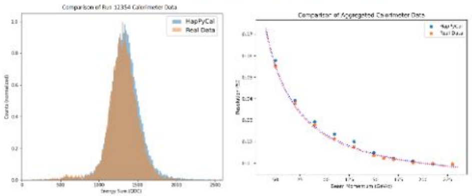


In the training process, **the generator repeatedly tries to "fool" the discriminator**. At first, the generator's outputs are recognized as fake every time, but by applying backpropagation algorithms, **the generator is able to "learn" from its mistakes** and gradually get better and better at producing "realistic" data. The discriminator is trained in tandem, creating a type of two-player minimax game.

On a more technical level, we use PyTorch to structure our generator as a **convolutional neural network** with three layers and a combination of ReLU and TanH activation functions. The discriminator is a standard **binary classification model** with a cross entropy loss function.

Results

After training our model on ~5% of the calorimeter data from each dataset over a target range of momenta, we **compare HapPyCal-generated data** (blue) with **real data** collected by the πM1 calorimeter (orange) below:



1. On the **left**, we focus on a *single* "run" (**fixed momentum:** 230 MeV/c) and show the distribution of total energy deposited by electron scattering events.
2. On the **right**, we look at *many* runs (**variable momentum:** 50-230 MeV/c) and show how the energy resolution (proportional to standard deviation) changes with momentum.

In both cases, the curve-fits for HapPyCal data and real data are incredibly close, suggesting that the generative modelling technique is **accurately "learning" the underlying distributions** of electron scattering data.

We note that in most of our analyses, HapPyCal's energy values tend to be greater than in real data. This can be attributed to the fact that GANs are usually susceptible to noisy outputs when the training data can't be normalized as in our case.

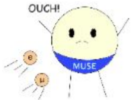
Conclusions / Further Work

The speed and accuracy of HapPyCal opens up some interesting opportunities for deeper analysis at MUSE. On the simulation side, analysis of GEANT4 models is now significantly faster. On the experimental side, we can also evaluate how detector responses vary over small periods to identity any drift in the data.

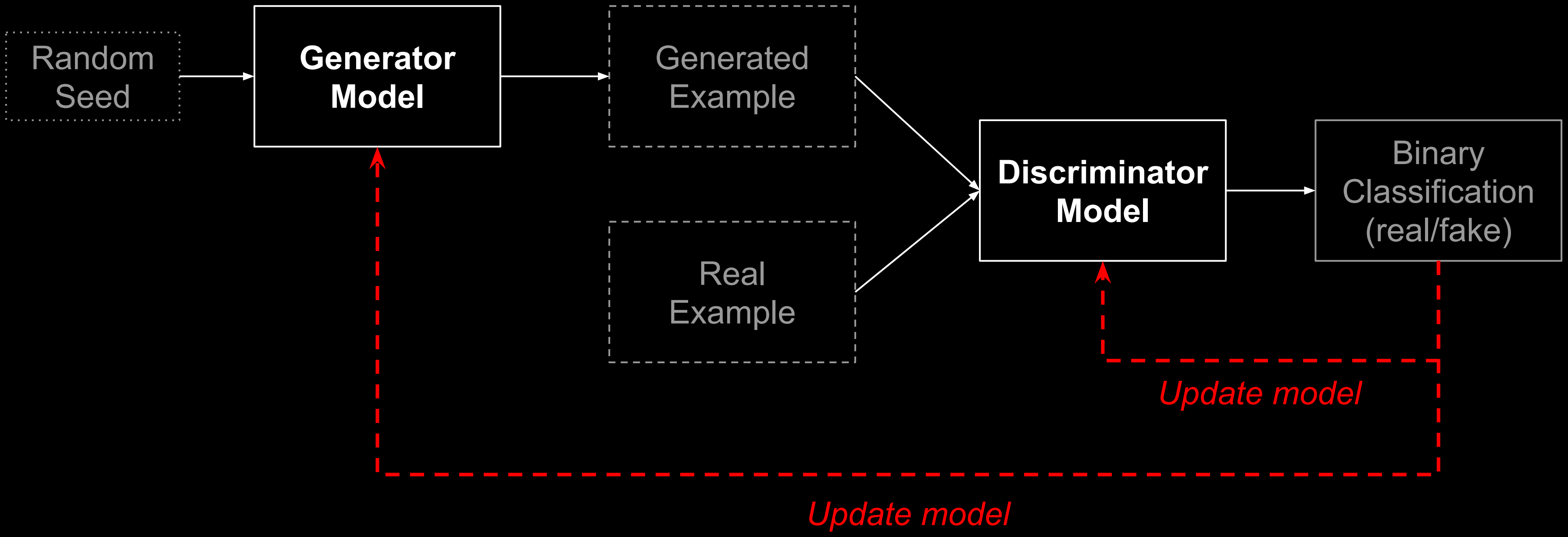
GANs are a powerful tool, but have many well known problems such noisy outputs and unstable/non convergent training. Future work with more cutting-edge techniques from machine learning literature such as Normalizing Flows and WGAN-GPs could offer even greater speed and accuracy.

Acknowledgements

This material is based upon work supported by Aresty and the National Science Foundation under NSF grant PHY-1913653. The MUSE experiment is supported by the Department of Energy, NSF, PSI, and the US-Israel Binational Science Foundation.



Generative Adversarial Network



HapPyCal: A Fast Calorimeter Simulation Framework for the πM1 Beam Line

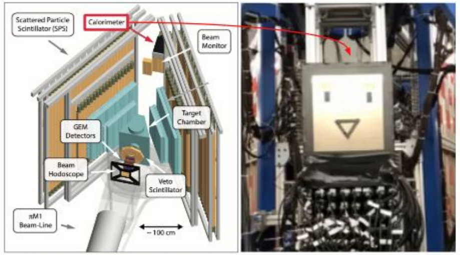
Rutgers Aresty Research Center for Undergraduates

Zain Eris Kamal; Advised by Ronald Gilman and Wan Lin
Department of Physics and Astronomy, Rutgers University, Piscataway NJ

The Proton Radius Puzzle

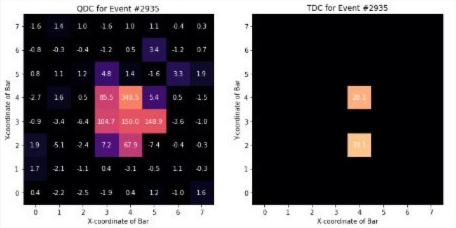
- In 2010, Pohl et al. measured the proton radius using *muonic* hydrogen spectroscopy and found it to be **5.6 σ smaller** than the accepted value determined through *electronic* hydrogen. It is still unclear how much of this difference results from interesting physics vs. experimental error.
- The **MUon Scattering Experiment (MUSE)** performs simultaneous muon and electron scattering off of a liquid hydrogen target with the goal of measuring the proton radius with high-precision. This will also provide a test of lepton universality and the possibility of two-photon exchange by switching between beam polarities.

πM1 Calorimeter



The πM1 beam line at the Paul Scherrer Institute has an electromagnetic **calorimeter** (pictured above) which measures the ***energy and timing* of particles moving along the beam line**. When high-energy muons/pions/electrons enter the calorimeter, they initiate a "particle shower", a cascade of lower-energy secondary particles which deposit their energy into the calorimeter.

The calorimeter is an 8 by 8 array of independently read out crystals, which are monitored in 2 ways: QDC (Charge-to-Digital Converter) and TDC (Time-to-Digital Converter). We can visualize the data collected by the calorimeter for a single event in the colormaps below, with the "hotter" pixels representing particle impact sites.



Abstract

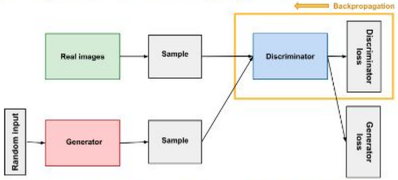
In high energy physics, detailed and accurate simulations of detectors (such as calorimeters) are the **single most expensive element of computational pipelines**. This is because most analyses require aggregated distributions of detector data as opposed to "realistic" data of individual events, creating heavy demand for fast simulation frameworks.

Traditionally ***GEANT4*** software is used to simulate detector responses, but this can take **minutes per individual scattering event** (of which millions are required) and is prone to systematic deviations from experimental data.

Thus we develop ***HapPyCal***, a fast detector simulation framework which can accurately reproduce calorimeter energy scans significantly faster than previous methods. We adapt a machine learning architecture called "Generative Adversarial Networks" and train it with data collected from experimental runs. Through analyses of energy scans from electron scattering events, we find that our model can **accurately reconstruct a larger distribution of data** with just **5%** of the original dataset.

Methodology

Generative Adversarial Networks (GANs) are a type of deep generative model characterized by two key components: a generator network and a discriminator network. The generator's goal is to create realistic data (in this case, calorimeter energy scans), while the discriminator's goal is to be able to tell if some given data is real or created by the generator.

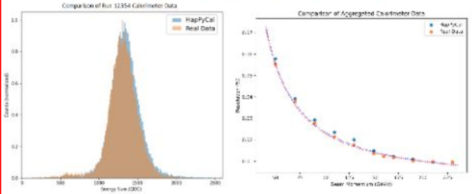


In the training process, **the generator repeatedly tries to "fool" the discriminator**. At first, the generator's outputs are recognized as fake every time, but by applying backpropagation algorithms, **the generator is able to "learn" from its mistakes** and gradually get better and better at producing "realistic" data. The discriminator is trained in tandem, creating a type of two-player minimax game.

On a more technical level, we use PyTorch to structure our generator as a **convolutional neural network** with three layers and a combination of ReLU and TanH activation functions. The discriminator is a standard **binary classification model** with a cross entropy loss function.

Results

After training our model on ~5% of the calorimeter data from each dataset over a target range of momenta, we **compare HapPyCal-generated data** (blue) with **real data** collected by the πM1 calorimeter (orange) below:



1. On the **left**, we focus on a *single* "run" (**fixed momentum:** 230 MeV/c) and show the distribution of total energy deposited by electron scattering events.
2. On the **right**, we look at *many* runs (**variable momentum:** 50-230 MeV/c) and show how the energy resolution (proportional to standard deviation) changes with momentum.

In both cases, the curve-fits for HapPyCal data and real data are incredibly close, suggesting that the generative modelling technique is **accurately "learning" the underlying distributions** of electron scattering data.

We note that in most of our analyses, HapPyCal's energy values tend to be greater than in real data. This can be attributed to the fact that GANs are usually susceptible to noisy outputs when the training data can't be normalized as in our case.

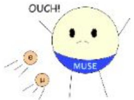
Conclusions / Further Work

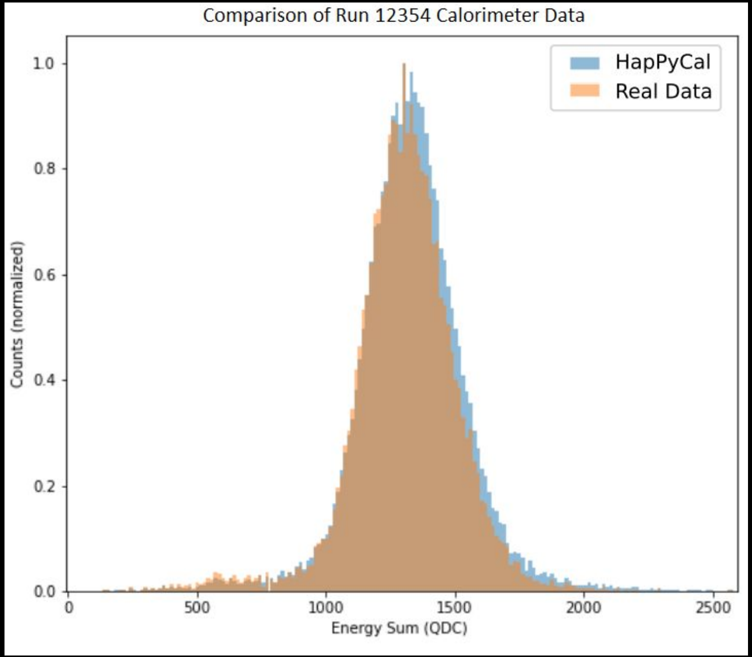
The speed and accuracy of HapPyCal opens up some interesting opportunities for deeper analysis at MUSE. On the simulation side, analysis of GEANT4 models is now significantly faster. On the experimental side, we can also evaluate how detector responses vary over small periods to identity any drift in the data.

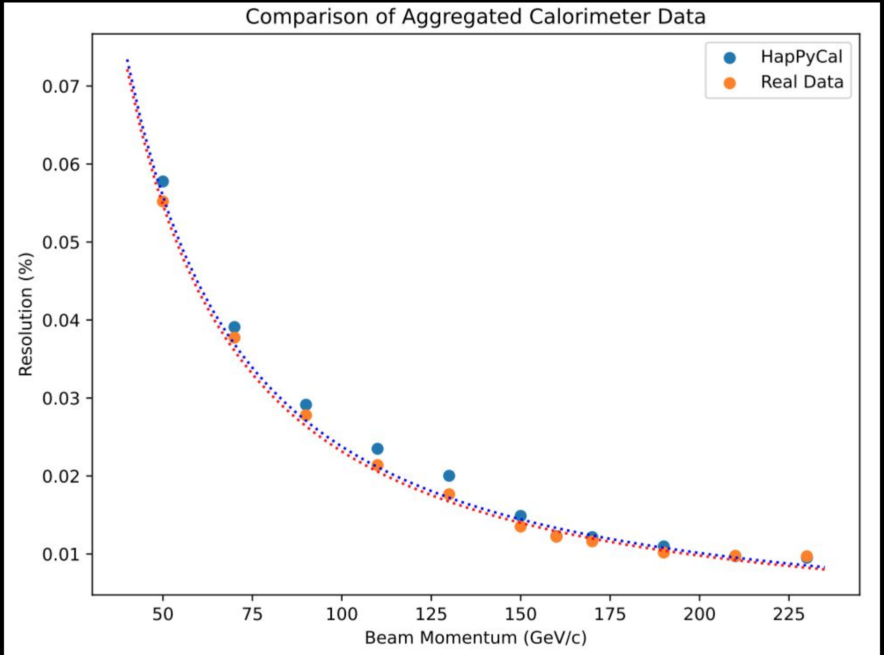
GANs are a powerful tool, but have many well known problems such noisy outputs and unstable/non convergent training. Future work with more cutting-edge techniques from machine learning literature such as Normalizing Flows and WGAN-GPs could offer even greater speed and accuracy.

Acknowledgements

This material is based upon work supported by Aresty and the National Science Foundation under NSF grant PHY-1913653. The MUSE experiment is supported by the Department of Energy, NSF, PSI, and the US-Israel Binational Science Foundation.




Comparison of Run 12354 Calorimeter Data
HapPyCal
Real Data
Counts (normalized)
Energy Sum (QDC)
0.0
0.2
0.4
0.6
0.8
1.0
0
500
1000
1500
2000
2500


Comparison of Aggregated Calorimeter Data
HapPyCal
Real Data
Resolution (%)
Beam Momentum (GeV/c)
0.01
0.02
0.03
0.04
0.05
0.06
0.07
50
75
100
125
150
175
200
225

HapPyCal: A Fast Calorimeter Simulation Framework for the πM1 Beam Line

RUTGERS Aresty Research Center for Undergraduates

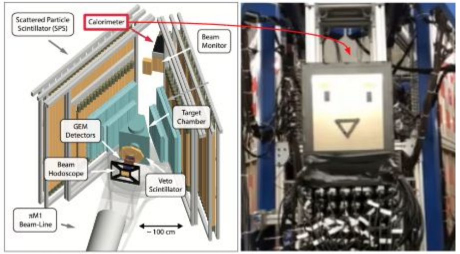
Zain Eris Kamal; Advised by Ronald Gilman and Wan Lin

Department of Physics and Astronomy, Rutgers University, Piscataway NJ

The Proton Radius Puzzle

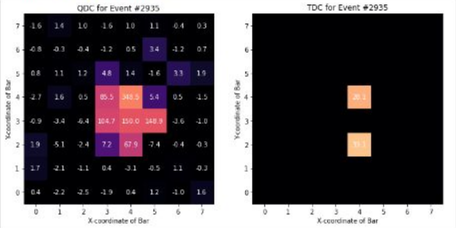
- In 2010, Pohl et al. measured the proton radius using *muonic* hydrogen spectroscopy and found it to be **5.6 σ smaller** than the accepted value determined through *electronic* hydrogen. It is still unclear how much of this difference results from interesting physics vs. experimental error.
- The **MUon Scattering Experiment (MUSE)** performs simultaneous muon and electron scattering off of a liquid hydrogen target with the goal of measuring the proton radius with high-precision. This will also provide a test of lepton universality and the possibility of two-photon exchange by switching between beam polarities.

πM1 Calorimeter



The πM1 beam line at the Paul Scherrer Institute has an electromagnetic **calorimeter** (pictured above) which measures the ***energy and timing* of particles moving along the beam line**. When high-energy muons/pions/electrons enter the calorimeter, they initiate a "particle shower", a cascade of lower-energy secondary particles which deposit their energy into the calorimeter.

The calorimeter is an 8 by 8 array of independently read out crystals, which are monitored in 2 ways: QDC (Charge-to-Digital Converter) and TDC (Time-to-Digital Converter). We can visualize the data collected by the calorimeter for a single event in the colormaps below, with the "hotter" pixels representing particle impact sites.



Abstract

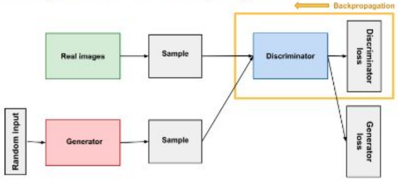
In high energy physics, detailed and accurate simulations of detectors (such as calorimeters) are the **single most expensive element of computational pipelines**. This is because most analyses require aggregated distributions of detector data as opposed to "realistic" data of individual events, creating heavy demand for fast simulation frameworks.

Traditionally ***GEANT4*** software is used to simulate detector responses, but this can take **minutes per individual scattering event** (of which millions are required) and is prone to systematic deviations from experimental data.

Thus we develop ***HapPyCal***, a fast detector simulation framework which can accurately reproduce calorimeter energy scans significantly faster than previous methods. We adapt a machine learning architecture called "Generative Adversarial Networks" and train it with data collected from experimental runs. Through analyses of energy scans from electron scattering events, we find that our model can **accurately reconstruct a larger distribution of data** with just **5%** of the original dataset.

Methodology

Generative Adversarial Networks (GANs) are a type of deep generative model characterized by two key components: a generator network and a discriminator network. The generator's goal is to create realistic data (in this case, calorimeter energy scans), while the discriminator's goal is to be able to tell if some given data is real or created by the generator.

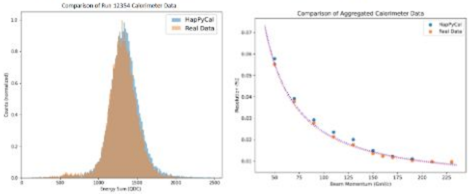


In the training process, **the generator repeatedly tries to "fool" the discriminator**. At first, the generator's outputs are recognized as fake every time, but by applying backpropagation algorithms, **the generator is able to "learn" from its mistakes** and gradually get better and better at producing "realistic" data. The discriminator is trained in tandem, creating a type of two-player minimax game.

On a more technical level, we use PyTorch to structure our generator as a **convolutional neural network** with three layers and a combination of ReLU and TanH activation functions. The discriminator is a standard **binary classification model** with a cross entropy loss function.

Results

After training our model on ~5% of the calorimeter data from each dataset over a target range of momenta, we **compare HapPyCal-generated data** (blue) with **real data** collected by the πM1 calorimeter (orange) below:



1. On the **left**, we focus on a *single* "run" (**fixed momentum:** 230 MeV/c) and show the distribution of total energy deposited by electron scattering events.
2. On the **right**, we look at *many* runs (**variable momentum:** 50-230 MeV/c) and show how the energy resolution (proportional to standard deviation) changes with momentum.

In both cases, the curve-fits for HapPyCal data and real data are incredibly close, suggesting that the generative modelling technique is **accurately "learning" the underlying distributions** of electron scattering data.

We note that in most of our analyses, HapPyCal's energy values tend to be greater than in real data. This can be attributed to the fact that GANs are usually susceptible to noisy outputs when the training data can't be normalized as in our case.

Conclusions / Further Work

The speed and accuracy of HapPyCal opens up some interesting opportunities for deeper analysis at MUSE. Notably, Monte Carlo analyses are more feasible, GEANT4 models can be evaluated much faster, and we can quantify how detector responses vary over small periods of time to identity any drift in the data.

GANs are a powerful tool, but have many well known problems such noisy outputs and unstable/non convergent training. Future work with more cutting-edge techniques from machine learning literature such as Normalizing Flows and WGAN-GPs could offer even greater speed and accuracy.

Acknowledgements

This material is based upon work supported by Aresty and the National Science Foundation under NSF grant PHY-1913653. The MUSE experiment is supported by the Department of Energy, NSF, PSI, and the US-Israel Binational Science Foundation.