Project Title:

Analysis & Dark Matter Physics Simulation for the Dark Photon

Team Members:

- Nikhil Chaba, nchaba2021@my.fit.edu
- Samuel Rock, srock2023@my.fit.edu
- Jacob Woods, jwoods2022@my.fit.edu

Faculty Advisor:

Pietro Iapozzuto, piapozzuto2015@my.fit.edu

Clients:

- Dr. Marcus Hohlmann - Physics Professor
- Pietro Iapozzuto - Graduate Student with Dr. Hohlman
- Other Graduate Physics Students

Dates of Meetings:

- 1/15/26
- 1/20/26
- 1/22/26

Goal:

Creating a software program that analyzes Dark Matter Photon searches for the EIC ePIC detector.

Motivation:

Currently there is no streamline or effective way to test Beyond Standard model Dark matter processes in the current EIC epic detector simulation. The client would like to be able to have software that can use a dark matter model, visualize it in the current EIC visualization, locate the vertices of the interaction, and obtain other physics quantities needed.

Approach and Key Features:

- **Visualize the decay products of the dark matter interaction in the EIC Geometry.** Focuses on forming a clear picture of how particles created in a dark matter interaction move through the Electron–Ion Collider detector. By tracking where these particles travel and which detector regions they pass through, researchers can understand what the event should look like in the experiment.

This helps ensure that the detector is being used effectively to observe possible dark matter-related signals.

- **Find the most probable dark matter signal (invariant mass, pseudorapidity (eta) , location).** Goal is to identify the key features that make a dark matter event most likely. This includes calculating the invariant mass of the detected particles, measuring their pseudorapidity (η) to understand how forward or central they are, and determining where in the detector they appear. These properties help narrow down the phase space where a true dark matter signal is expected to stand out.
- **Correctly perform background subtraction for competing processes (ie Pions).** This step is about removing signals from known Standard Model processes that can mimic dark matter events. Particles like pions are produced frequently and can look similar to the desired signal. By estimating how often these background processes occur and subtracting their contribution, researchers can isolate what remains and reduce false positives.
- **Correct signal matching for the dark matter e+ e-.** The final step ensures that the detected electron and positron are correctly paired as coming from the same dark matter interaction. This involves matching their tracks and energies so they form a consistent event. Proper signal matching prevents misidentifying unrelated particles as a dark matter decay and improves the reliability of the final results

Novel features/functionalities:

- Visualization of the dark matter tracks that is able to be shown and explained to a customer or nontechnical person.
- Background subtraction of user specific dark matter process allowing for matching of dark matter between simulation and real data.

Algorithms and Tools:

- Github - Host as our website and version control
- Mad Graph - CLI Modeling Software
- HepMC - Object oriented C++ event record in High Energy physics
- ROOT - Data analysis framework to create histograms, can work as a python library
- EIC Shell - Containerized software environment for the Electron-Ion Collider
- Python - Programming language
- Fortran Compiler - To run “launch” in Mad Graph

Technical Challenges:

- Plan to use the Github IO page for our website, but do not know much HTML
- It will be necessary to use Mad Graph, HepMC, ROOT, and EIC and the processes between them to create diagrams of particles, but we will need to learn completely from scratch on how to use these
- Also necessary to use ROOT and identify Dark photon and decay products, will need to understand how to use this software

Milestone 1 (Feb 23):

- Learn and understand how to use Mad Graph, HepMC, and ROOT
- Read HepMC files
- Use ROOT to identify particle id numbers ie 1023 is dark photon candidate as a parent particle and its decays (children) and graph physics related quantities ie Momentum of scattered particle
- Install EiC RECO and Use EIC Geometry, NPSIM to output reconstruction root file
- Use ROOT to graph signals of initial proton and electron, the scattered electron and decay products at the madgraph generator level , and compare it to the EIC reconstruction level
- Produce EIC visualization using inner detector obtaining dark matter decay product e- e+ tracks
- Using HepMC file data, produce a Python script that works along with ROOT to produce histograms of scattered electron energy
- Create Requirement Document
- Create Design Document
- Create Test Plan

Milestone 2 (Mar 30):

- Complete graph of background subtracted signal of dark matter at the end of the EIC pipeline
- Explore 10-20 different invariant masses, running them through simulation pipeline, visualization to identify most probable dark matter invariant mass, and incorporating the background subtracted

Milestone 3 (Apr 20):

- Streamline the process
- Get cross section comparison at generator level, at reconstruction level, Pseudorapidity (Eta), momentum transfer for each invariant mass that is put in pipeline and a visualization of tracks

Task matrix for Milestone 1:

Task	Nikhil	Sam	Jacob

Learn and understand how to use Mad Graph, HepMC, and ROOT	1/3	1/3	1/3
Start reading HEPMC files	1/3	1/3	1/3
Use ROOT to identify particle id numbers ie 1023 is dark photon candidate as a parent particle and its decays (children) and graph physics related quantities ie Momentum of scattered particle	1/3	1/3	1/3
Install EiC RECO and Use EIC Geometry, NPSIM to output reconstruction root file	1/3	1/3	1/3
Use ROOT to graph signals of initial proton and electron, the scattered electron and decay products at the madgraph generator level, and compare it to the EIC reconstruction level	1/3	1/3	1/3
Produce EIC visualization using inner detector obtaining dark matter decay product e- e+ tracks	1/3	1/3	1/3
Using HepMC file data, produce a Python script that works along with ROOT to produce histograms of scattered electron energy	1/3	1/3	1/3
Requirement Document	1/3 *	1/3	1/3
Design Document	1/3	1/3 *	1/3
Test Plan	1/3	1/3	1/3 *