

A Simulation of High Energy Physics Software Related to Dark Photons

Ji Young CHOI · Kyung Kwang JOO*

Institute for Universe & Elementary Particles, Department of Physics,
Chonnam National University, Gwangju 61186, Korea

(Received 27 March 2018 : revised 1 June 2018 : accepted 1 June 2018)

This paper aims to describe a minimum methodology and the minimum knowledge for end users to assist in performing the underlying views and computer simulations. For this purpose, a basic analysis of a high energy physics software simulation related to dark photons was performed as an example. Dark matter is one of the most important research subjects in modern particle physics. Weakly interacting massive particles, dark higgs bosons, axions, sterile neutrinos, and dark photons are candidate particles for dark matter. The general method is to model dark matter theory and then to test it through experiment. During the testing phase, high energy physics software packages are used for various verification and assisting measurements.

PACS numbers: 95.35.+d, 07.05.Tp

Keywords: High energy physics software, Dark photon, Physics simulation, Data analysis

암흑광자 관련 고에너지 물리 소프트웨어 전산모사 연구

최지영 · 주경광*

전남대학교 물리학과, 우주소립자 연구소, 광주 61186, 대한민국

(2018년 3월 27일 받음, 2018년 6월 1일 수정본 받음, 2018년 6월 1일 게재 확정)

본 논문은 고에너지 물리 소프트웨어관련 전산 모사를 수행할 때 밑바탕을 이루고 있는 관점 및 최소한의 필요한 방안과 지식을 언급하고자 한다. 이를 위해 하나의 예제로 암흑 광자 관련 기초 분석을 수행하였다. 암흑 물질은 현재 입자물리학에서 매우 중요한 연구 과제중의 하나인데, 약하게 상호작용하는 무거운 입자, 암흑 힉스, 엑시온, 비활성 중성미자, 암흑 광자 등이 후보로 제안되었다. 일반적인 연구 방법은 암흑 물질의 이론이 적용 된 모형을 만든 후 실험 통해서 그 이론을 검증하는 방법을 택하고 있다. 테스트하는 단계에서 여러 가지 검증 및 보조 수단의 일환으로 고에너지 물리 소프트웨어를 활용한다.

PACS numbers: 95.35.+d, 07.05.Tp

Keywords: 고에너지 물리 소프트웨어, 암흑광자, 전산모사, 데이터 분석

*E-mail: kkjoo@chonnam.ac.kr



I. 서론

일반적으로 고에너지 물리 분야의 실 사용자(end user)는 물리 사건 모사(physics generator), 검출기 전산모사(detector simulation), 사건 재구성(trigger and event reconstruction), 데이터 분석 및 해석(data analysis and interpretation) 단계에서 고에너지 물리(high energy physics, HEP) 소프트웨어를 경험하고 사용한다 [1]. 이러한 HEP 소프트웨어가 구동되는 운영체제(operating system) 환경은 계산(computing) 환경 관리 편의성 때문에 UNIX 계열이 선호되고 실 사용자는 그에 맞는 전산 지식 및 훈련이 필요하다. 실 사용자 수준에서 운영체제에 대한 이해의 핵심은 운영체제의 구조 및 명령어 해석기(shell)를 얼마나 잘 이해하고 사용하는 것이다. 특히 셸 프로그래밍이 익숙하지 않은 실 사용자가 많이 범하는 실수의 대부분은 셸 변수, 환경변수 이해 부족이다 [2]. HEP 소프트웨어 설치하는 소프트웨어 개발자 그룹 및 실험 그룹에서 해당 개발 환경에서 작성한 설치 셸 스크립트 배포한 것을 실행시키면 되는데, 셸 프로그래밍이 익숙하지 않으면 설치 오류가 발생할 경우가 많다. 또한 HEP 소프트웨어들은 다양한 컴퓨터 언어로 작성되었기 때문에 실 사용자는 해당된 컴퓨터 언어의 문법을 익히면서 프로그래밍 기술을 숙련해야한다 [3–16]. 이때 실 사용자는 수많은 오류들을 접하고 해결하는 과정에서 여러 시행착오를 겪게 되는데 많은 경우 컴퓨터 구조 및 언어 인식 체계에 대한 경험 부족에서 비롯된다. 따라서 고에너지 물리 연구를 보다 빠르게 수행하기 위해 운영체제, 컴파일러, 프로그래밍 언어론 지식 등 구체적인 컴퓨터 언어 문법을 알면 많은 도움이 된다 [17–20]. 그러나 프로그래밍 기술 숙련 과정은 연구자에게 큰 부담을 준다. 신입 연구자의 경우에는 HEP 및 범용 소프트웨어 개발자 커뮤니티의 전문가들이 지도하는 교육 목적의 해커톤(hackathon)에 참여하거나, 해당 커뮤니티에 보고된 표준 개발 환경에서의 실 사용자 오류 보고서 색인 검색 서비스(database)를 이용할 수도 있으며 해당 커뮤니티에서 업로드된 소스 코드 예제를 사용해 빌드(build) 및 실행 후 결과 분석 방법을 선택할 수도 있다. 본 논문은 HEP 소프트웨어 시뮬레이션을 하는데 도움을 줄 수 있는 최소한의 방안 및 지식을 기술하고자 한다. 이를 위한 하나의 예제로서 암흑물질 후보 중에서 암흑 광자 관련 기초적인 연구 분석을 하였다.

입자 물리학의 해결되지 않은 여러 질문 중의 하나는 암흑 물질에 대한 우주론적 기원, 구성 요소, 여러 가지 상호작용 등의 근본적인 성질을 밝히는 것이다. 암흑 물질의 존재 증거는 은하에 대한 우주 배경 복사 모양, 회전 속도, 중력 렌즈 효과 같은 우주론적, 천체 물리학적 관측에서

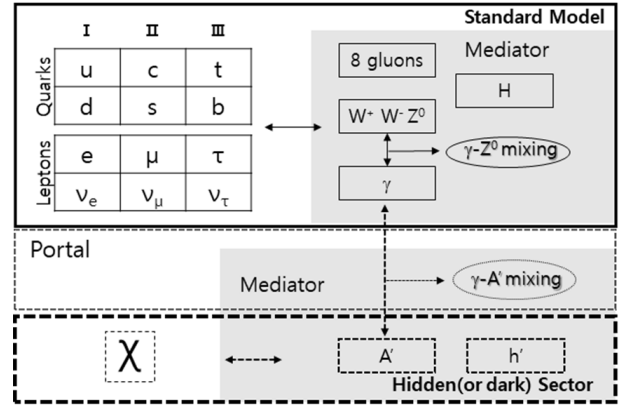


Fig. 1. A kinetic mixing model for dark photon.

유래되었다. 일반적으로 암흑물질에 대한 연구는 이론이 적용된 모델을 만들고 실험을 통해서 그 이론을 검증하는 방법을 택하고 있다. 현재, 각광을 받는 암흑 물질 모델에 대한 분류는 암흑 물질 후보군의 질량 및 구성 입자의 종류로 크게 나눌 수 있다. 암흑 광자 모형은 동역학적 섞임(kinetic mixing) 모형, 중입자수(baryon number)와 경입자수(lepton number) 차이에 관한 대칭 및 게이지 입자 존재를 가정하는 모형(B-L model), 경입자들에 대해서만 게이지 대칭성 존재 모델(leptophilic gauge boson model), 경입자의 각 세대별로 다른 결합 상수를 갖는 모델(flavor or lepton non-universal coupling) 등이 있다 [21–28]. Fig. 1은 가장 단순하게 정의되어 있는 동역학적 섞임 모형을 나타낸다. 이 논문은 이 모델을 기반으로 소프트웨어 분석을 시도하였는데 표준 모형의 전자기 약력 이론의($\gamma - Z^0$) 섞임과 유사하게 암흑 물질(암흑 광자, A')과 표준 모형 물질(광자, γ) 사이의 섞임 있다고 가정한다 [21]. 암흑광자가 표준 모형의 물질로 붕괴하는 경우를 보이는 암흑광자 붕괴(visible dark photon decay)라 한다.

II. 전산모사를 위한 소프트웨어

1. 물리 이벤트 생성기(event generator)

물리 이벤트를 생성하기 위해 몬테카를로(Monte Carlo, MC) 사건 생성기를 이용하는데, 고에너지 실험의 데이터들을 해석하고 이해하는데 핵심적인 역할을 제공한다. 고에너지 이론에 기초한 모델을 포함한 MC 사건 생성기들은 표준 HEP 라이브러리들을 이용해서 각각의 파인만 다이어그램에 해당하는 위상 및 확률 진폭을 계산하고 그 결과를 부여받아 가상의 입자(MC particles)들을 생성한다 [29]. 또한 이론에 기초한 해당 모델들이 다르므로 사건 생성기들의

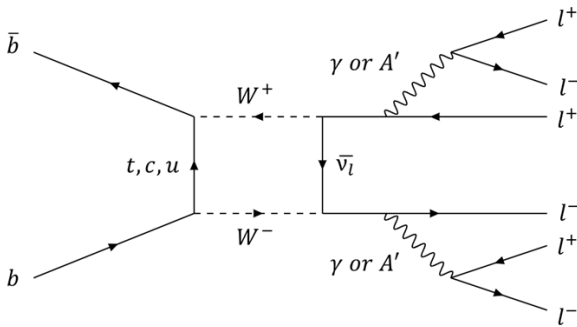


Fig. 2. An example of a Feynman diagram where B meson decay into six leptons (through dark photons).

한계와 목적을 잘 알아야 한다. Particle Data Group에서 발간한 입자 데이터 테이블 (particle data table) 및 입자 붕괴 모형 목록 (particle decay model list)을 이용할 수 있는데, 실 사용자가 원하는 암흑물질 관련 입자와 모형을 형식에 맞게 추가하고 수정한다. 본 연구의 MC 사건 주 생성기는 EvtGen을 사용하였고, 여기서 암흑광자 관련 분석을 위해 PYTHIA8.2와 광자 방출 보정을 위해 PHOTOS를 이용하였다. 그리고 충돌 환경은 빔 에너지가 비대칭인 전자-양전자가 충돌되어 질량중심 에너지가 11GeV인 Upsilon(4S)가 생성되고 붕괴되어 B 중간자가 만들어진 후 암흑광자를 통한 보이는 붕괴 (visible dark photon decay)로 설정하였다 [30–32]. 암흑광자의 질량은 $1 \text{ GeV}/c^2$ 이고 넓이 (width)는 1 MeV 이며 갈래비 (branching ratio)는 0.15×10^{-10} 을 사용하였다. 최종적으로는 경입자 (뮤온) 쌍이 만들어지는 채널을 분석하였는데 이와 관련된 파이만 다이어그램은 Fig. 2에 있다. Fig. 3은 MC 사건 생성기를 사용하여 암흑 광자가 경입자 쌍으로 붕괴되는 하나의 이벤트를 생성한 결과를 보여주며 각 색깔은 여러 가지 입자를 나타낸다. 사전에 정의된 검출기 기하 중심점과 사건 생성 후 출력된 상호작용 중심점 (interaction point, IP)을 정렬한 다음 시각화 소프트웨어로 처리한 결과이다.

2. 검출기 전산 모사 (detector simulation)

물리 사건 생성과 다르게 검출기 전산 모사 소프트웨어는 주로 Geant4 소프트웨어가 핵심적인 역할을 한다. 검출기 전산 모사에서는 물리 사건 생성기에서 출력된 입자 (MC particle)를 검출기 전산 모사 프로그램에서 자체적으로 호출한 입자 (Geant 4 particle)에 상속 및 일치시킨 생성 입자 (matched MC particle)와 검출기 물질 (Geant4 material)간의 상호작용 및 수송 현상 (transport)을 모사한다 [33]. 이 단계에서 실 사용자는 MC particle ID와

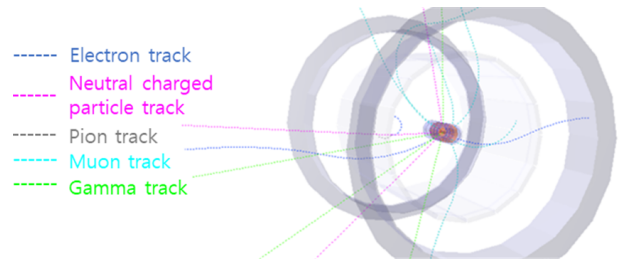


Fig. 3. (Color online) Example for several events, including dark photon decay to ordinary matter of SM, produced by using MC event generator around IP. Neutral charged particle track (magenta color), electron track (blue color), muon track (sky blue color), and pion track (gray color) are drawn.

Geant4 particle ID를 일치 (또는 상속) 시켜 주는 것이 중요하다. 정합 (match) 여부에 따라, 입자들의 단면적, 검출기에서 모 입자들 궤적 및 딸 입자들 궤적, 강입자 및 전자기 물리과정에 따른 부딪힌 경우 (hit, score)가 달라질 수 있다. 실 사용자는 자기장을 구현한 모듈 인식관련 오류들은 해결하기가 어렵기 때문에 검출기 전산 모사에서 주로 전자기 물리를 정확하게 부여하고 묘사하는 것이 필요하다. Geant4에서 입자 수송을 묘사하는데 병렬처리 및 자료구조가 효율적이거나 최적화 되지는 않았다. MC 입자 및 공명 입자들이 검출기 물질에 남긴 축적된 에너지, 궤적 등의 물리량을 계산하기 위해 이 단계에서 가장 많은 컴퓨터 자원을 요구하며 이런 점 때문에 오류를 인식하는 시간이 가장 길다. 이러한 병렬처리의 단점을 개선하기 위해 GeantV 프로젝트가 진행 중이다.

3. 사건 재구성 (software trigger and event reconstruction)

검출기 전산 모사에서 입자들은 검출기 물질과 상호작용 인하여 여러 가지 물리량을 발생시킨다. 예를 들어 에너지 저지능 (dE/dx), 채공 시간 (time of flight), 체렌코프 방사선 (Cherenkov radiation), 전이 방사선 (transition radiation), 운동량 및 위치 정보 같은 운동학적 제약치 (kinematics constraint) 등을 알 수 있고 검출기 물질에 축적된 물리량에 따라 입자의 종류를 구별 할 수 있다. 실 사용자는 오프라인 소프트웨어 및 온라인 검출기 하드웨어 상에서 트리거 (trigger), 궤적 추적 (tracking) 그리고 사건 재구성 알고리즘에 대한 전반적인 이해가 매우 중요하다. 본 연구는 암흑광자가 경입자 쌍으로 붕괴되기 때문에, 이 사건을 잘 재구성하기 위해서 하드웨어 트리거로는 실리콘 검출기 (silicon vertex detector), 전자기 열량계 (electro-magnetic calorimeter), 뮤온 검출기를 사용하였고 여기에 사건 형태

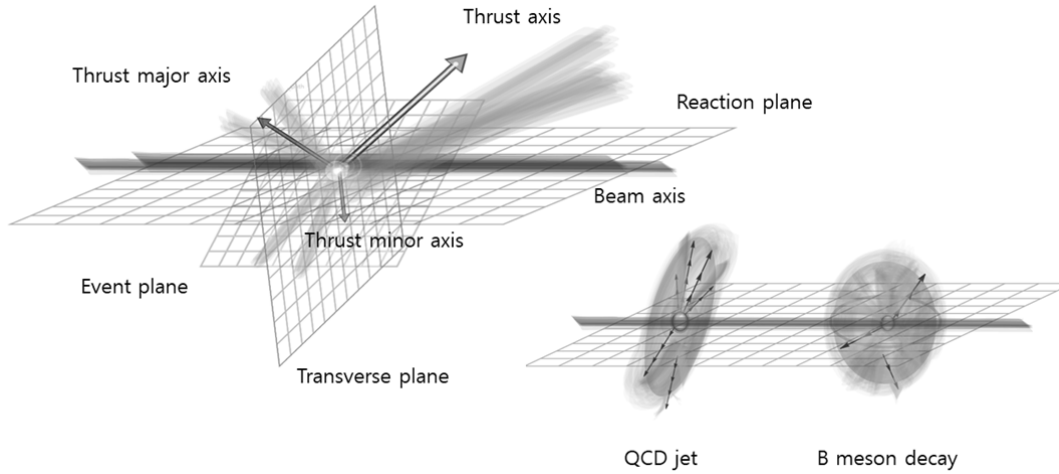


Fig. 4. Illustration of showing one of event shape variables. A thrust variable (T), which provides separation between QCD jets and B meson decay, is drawn.

변수 (event shape variable), 칼만 필터와 패턴 인식을 적용하였다 [4–37]. 사건의 재구성 (event reconstruction)은 실제 실험 데이터나 MC 생성 이벤트에 동일하게 사용할 수 있다. 먼저 MC를 이용해서 사건의 재구성을 하는 알고리즘을 만들거나 최적화할 수 있고, 이것을 다시 데이터에 적용할 수 있다. 사건 형태 변수는 전체적인 입자 빔 충돌 사건이 일어난 당시 입자 충돌, 산란, 생성 사건들 속에서 우세한 상호작용 종류, 위상공간에서 불변 산란 단면적 그리고 운동학적 구속 조건에 따라 실험 사용자가 원하는 신호인가 혹은 원하지 않는 배경사건인가를 판별하는 기준이 되는 변수이다. 사건 형태 변수는 빔 충돌 후 생성된 입자들에 대한 에너지 흐름의 기하학적 특성을 반영한다. Fig. 4의 좌측 상단 각 입자의 운동량 및 운동량을 이용한 사건 재구성 변수 중의 하나인 스러스트 (Thrust, T) 변수 분포를 나타내는 하나의 예이다. 만약 빔 에너지가 비대칭이면 실험실계 기준으로 빔 충돌 후 생성되는 강입자(hadron)들의 질량 중심계는 빔 축 방향으로 밀려나가게 (boost) 된다. 빔 충돌 후 최종 강입자들(final stated hadrons)의 운동량을 측정할 수 있다면 정규화(normalized)된 질량 중심계가 특정한 방향을 나타내는 단위 벡터가 존재하며 스러스트 축(thrust axis) 벡터는 정규화된 질량 중심계에서 가장 활동적인(energetic) 운동량 방향으로 택한다. 스러스트 주축(thrust major axis) 벡터 및 스러스트 단축(thrust minor axis) 벡터는 스러스트 축 벡터 및 서로에 대해서 직교(orthogonal)하며 스러스트 축 방향을 제외한 정규화된 중심 질량계의 각 방향에서 활동적인 에너지 및 운동량을 지닌 입자들의 흐름 방향을 대표하는 벡터이다. 스러스트 축 방향과 빔 축 방향의 외적(cross product)한 벡터는 사건 평면(event plane)에 수직이어서 이 법선(normal) 벡터로

사건 평면을 정의할 수 있다. 사건 평면은 빔 충돌 후 생성된 입자들이 놓여 있는 평면이다. 반응 평면(reaction plane)은 빔 충돌 시 충돌이 이항(binary) 충돌이라고 가정하고 빔 축 방향에 횡단면(transverse plane)에서 각각의 빔의 중심은 연결한 선으로 빔들이 놓여 있는 평면이다. 스러스트는 어미 입자가 붕괴하여 여러 딸 입자들 생성되어 흩어지는 양상에 따라 운동량 분포 달라지는데 스러스트 변수 값이 $T \sim 1$ 이면 모 입자가 특정 방향으로 2개 양자 색소 역학 제트(QCD jet)로 붕괴하고, $T \sim 0.5$ 이면 모든 방향에 대해서 등방적인 형태로 붕괴되는 것을 나타내는 물리 변수로 이를 통해 사건의 형태를 구별할 수 있다 [34]. Fig. 4의 우측 하단의 좌측 그림은 전형적인 양자 색소 역학 제트 사건 형태이고 우측은 B 중간자(meson) 붕괴 사건 형태에서 운동량 분포를 보여준다.

패턴 인식은 개발자 및 실험 사용자가 원하는 기준에 따라 분류하는 알고리즘 중의 하나이며, 지도학습 기계 학습(supervised learning machine learning) 및 다변량(multivariate) 통계를 이용한다 [38–41]. 물리 변수들의 상관관계(correlation) 및 통계량(statistics)을 뽑아내기 위해서는 사건 재구성 단계에서 ROOT data framework analysis toolkit (ROOT), The Toolkit for Multivariate Data Analysis with ROOT (TMVA), The Toolkit for Data Modeling with ROOT (RooFit), RooFit과 ROOT에 배포된 통계 패키지들을 통합 프로젝트(Roostat) 등과 같은 회귀 또는 다변량 통계 분석 패키지 문법을 익혀야 한다 [42]. Fig. 5는 검출기 전산 모사 후 저장된 정보에 트리거 및 궤도 추적 소프트웨어 패키지를 이용한 결과를 시각화한 것이다. Fig. 5는 Fig. 3에서는 볼 수 없는 생성 입자(MC particle)들과 검출기 물질간의 상호작용 및 그 결과 축적된 물리량들을 볼 수 있다.

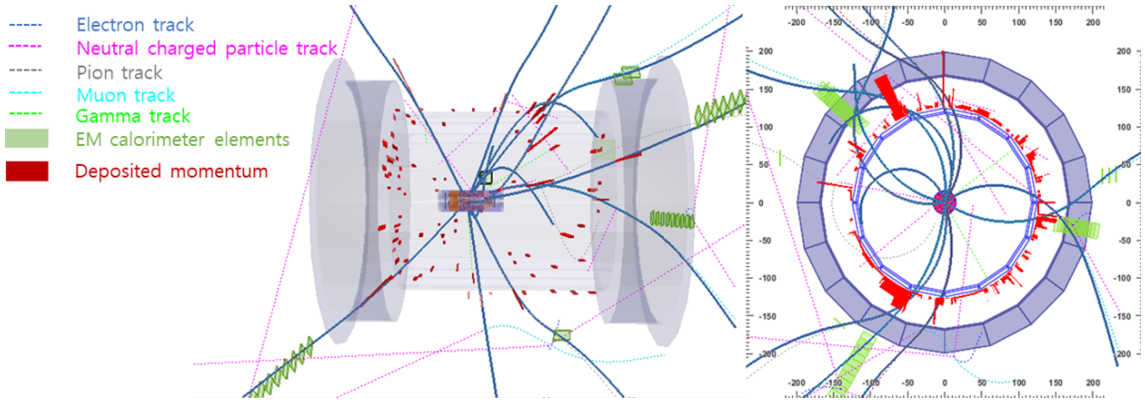


Fig. 5. (Color online) An example for events reconstruction around IP. Neutral charged particle track (magenta color), electron track (blue color), muon track (sky blue color), pion track (gray color), a response elements of electromagnetic (EM) calorimeter (green color box), and a deposited momentum on elements of detector material (red logo block histogram) are drawn.

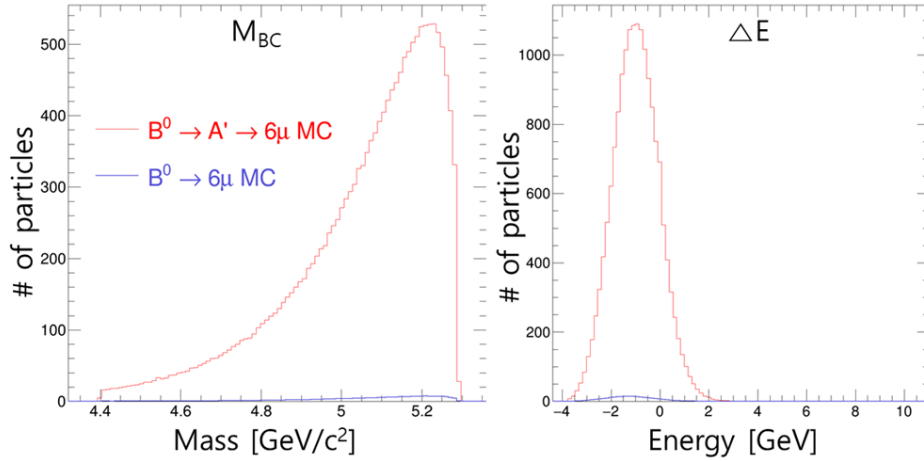


Fig. 6. (Color online) Example for visualization of raw data with using ROOT data framework analysis toolkit. (Left) beam-energy constrained mass of B meson $M_{bc} = \sqrt{E_{beam}^2 - p_B^2}$, where E_{beam} is the beam energy, and p_B is the momentum of the reconstructed B meson (Right) energy difference $\Delta E = E_B - E_{beam}$, where E_B is the energy of the reconstructed B meson. In this analysis channel, B meson decays into six leptons with or without dark photon.

4. 데이터 분석 및 해석 (data analysis and interpretation)

이 단계에서 실 사용자는 데이터 분석에서 요구되는 컴퓨터 자원량 정도에 관계없이 오프라인에서 많은 시간을 소비한다. 급속한 컴퓨터 하드웨어 발전 인하여 심층 학습 (deep learning) 기술이 적용된 패키지를 각 그룹에 맞게 최적화하여 적용하는 시도를 하고 있다. 주로 사건 재구성 단계에서 목적에 맞는 통합 통계 패키지를 이용해서 사용자가 입력한 데이터들을 가지고 최적의 매개변수 값을 찾는데, 이것이 기계학습에서 “학습”의 의미를 나타낸다. 고에너지 물리의 데이터 분석은 신호와 잡음의 분류 (classification)가 중요한데 여기에 다변량 해석 및 기계 학습 도구를 이용할 수 있다. 예로, 의사 결정 수행도 (decision tree), 무작

위 숲 (random forests), 부스트 의사 결정 수행도 (boost decision tree, BDT) 방법 등이 있는데, 데이터가 많이 존재해야 예측 및 해석이 용이하다 [38–41]. 이렇게 얻은 여러 매개변수들 값에 의해서 학습된 데이터와 학습 혹은 분류가 안 되어진 데이터 통해 사용자 원하는 모델과 통계적 추론을 교차 검증하거나 시각화를 통해 유의미한 해석 및 추론을 한다 [42–44]. 본 논문의 예제는 암흑광자가 생성되거나 혹은 생성되지 않은 후 바로 경입자 쌍으로 붕괴하는 채널을 분석하였기 때문에, 역으로 경입자 정보를 통해 B 중간자를 재구성할 수 있다. Fig. 6은 사건 재구성 후 출력된 데이터를 어떠한 정제 작업을 거치지 않고 미가공 (raw) 데이터를 ROOT를 통해 시각화 한 것이다. 재구성 후의 데이터는 저지능, 사건 형태 변수, 운동량 구속치 등의 추가 정보들이 저장되어 있고, Fig. 7은 미가공 데이터에서 원하지 않는

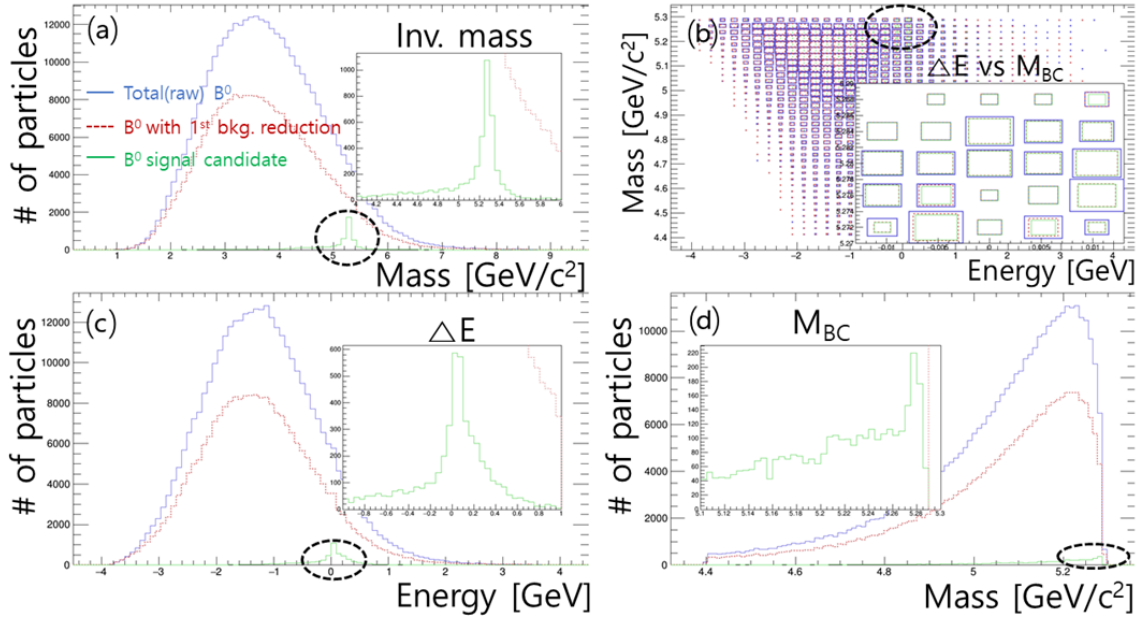


Fig. 7. (Color online) Example for visualization to raw data applied with user-customized cuts by using root. (a) invariant mass of B meson, (b) scatter plot with beam energy constraint mass versus energy difference of B meson, (c) energy difference of B meson, (d) beam energy constraint mass of B meson.

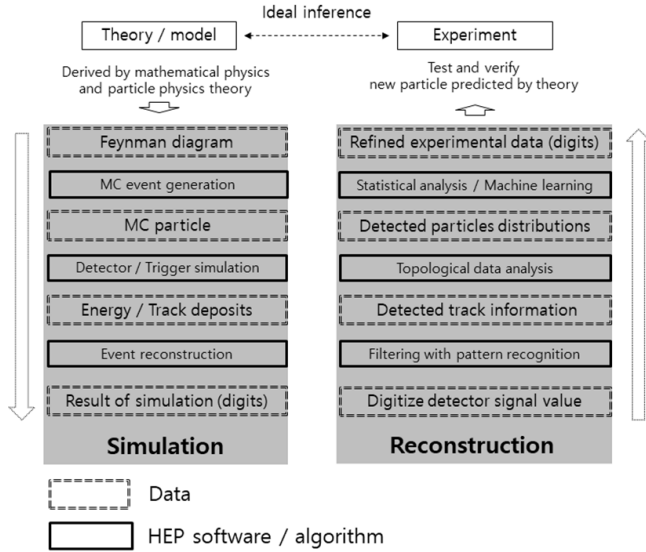


Fig. 8. Flow diagram to the view point of theory and experiment is drawn in particle physics research using HEP softwares. After the detector simulation and digitalization, the steps are the same for simulation and real data.

잡음 및 정보를 배제한 후 결과를 시각화한 것이다. 본 예제에서 주요 배경사건은 경입자로 잘못 재구성되거나 분류된 경우에서 비롯된다.

Fig. 8은 고에너지 이론과 실험에서 고에너지 물리용 소프트웨어의 정보 흐름을 보는 관점을 나타낸다. 고에너지 현상론과 이론 연구는 수리물리와 입자물리학의 방법론을

이용해서 이론 만들고 그것을 바탕으로 모델을 세운 후 한 전산 모사 프로그램을 만들거나 기존에 있는 프로그램을 수정해서 얻은 결과를 해석하여 모델의 예측과 실제 실험에서 얻은 정제된 결과를 비교 하면서 이론을 검증한다. 고에너지 실험 연구는 실험을 통해서 물리 현상을 관측하기 때문에 최대한 정확한 물리량을 얻기 위해 실제 참값에 근접한 이산화 (digitize) 신호 값을 저장하고 저장된 신호를 고에너지 소프트웨어 및 하드웨어를 이용해서 사건을 재구성하고 재구성된 사건에서 잡음과 신호를 분리 시켜 정제된 신호 데이터 얻고 이론 및 모델을 검증한다.

III. 요약

고에너지 물리 전산 모사는 이론 및 모델에 대한 이해가 밑바탕이 되어야 한다. 이와 함께 고에너지 물리 소프트웨어 및 사용 환경에 대해 요구 되어지는 지식의 깊이 또한 깊어지고 있다. 실 사용자를 위한 하나의 예제로 암흑 광자와 관련해 동역학적 섞임 모델 결과를 간단히 분석하였다. 암흑광자를 포함하는 MC 사건 생성기와 검출기 전산 모사를 통해 생성된 입자들이 검출기 물질과 상호작용 후 발생시킨 여러 입자나 물리량을 이벤트 디스플레이를 통해 시각적으로 나타내었다. 그리고 고에너지 물리 소프트웨어의 밑바탕을 이루는 관점 및 문제 해결에 유용한 최소한의 지식 및 방안에 대해 기술하였다.

감사의 글

이 논문은 2018년도 정부(과학기술정보통신부)의 재원으로 한국연구재단의 기초연구사업 (NRF-2016K1A3A7A09005608) 지원을 받아 수행된 연구입니다.

REFERENCES

- [1] HEP Software Foundation homepage, <http://hepsoftwarefoundation.org> (accessed Mar., 15, 2018).
- [2] R. Blum and C. Bresnahan, *Linux Command Line and Shell Scripting Bible* (Wiley, 2015).
- [3] S. J. Chapman, *Fortran for Scientists and Engineers* (McGraw-Hill, New York, 2017).
- [4] M. Metcalf, J. Reid and M. Cohen, *Modern Fortran Explained* (Oxford University Press, New York, 2011).
- [5] N. S. Clerman and W. Spector, *Modern Fortran Style and Usage* (Cambridge University Press, 2012).
- [6] S. Prata, *C Primer Plus* (Sams Publishing, 2004).
- [7] S. Prata, *C++ Primer Plus* (Sams Publishing, 2011).
- [8] S. Meyers, *Effective Modern C++* (O'Reilly, 2014).
- [9] I. Horton, *Using the C++ Standard Template Libraries* (Apress, 2015).
- [10] P. Gottschling, *Discovering Modern C++* (Addison-Wesley, 2016).
- [11] D. Beazley and B. K. Jones, *Python Cookbook* (O'Reilly, 2013).
- [12] L. Ramalho, *Fluent Python* (O'Reilly, 2015).
- [13] B. Slatkin, *Effective Python* (Addison-Wesley, 2015).
- [14] K. Beck, *Implementation Patterns* (Addison-Wesley, 2007).
- [15] J. Bloch, *Effective Java* (Addison-Wesley, 2017).
- [16] H. Schildt, *Java: A Beginner's Guide* (McGraw-Hill, 2017).
- [17] D. A. Patterson and J. L. Hennessy, *Computer Organization and Design* (Morgan Kaufmann, 2013).
- [18] A. B. Tucker and R. E. Noonan, *Programming Languages* (McGraw-Hill, 2008).
- [19] A. Silberschatz, P. B. Galvin and G. Gagne, *Operating System Concepts* (Wiley, 2013).
- [20] A. V. Aho et al., *Compilers: Principles, Techniques, and Tools* (Addison Wesley, 2006).
- [21] B. Holdom, *Phys. Lett. B* **166**, 196 (1986).
- [22] S. Weinberg, *Phys. Rev. Lett.* **43**, 1566 (1979).
- [23] H. Georgi, H. R Quinn and S. Weinberg, *Phys. Rev Lett.* **33**, 451 (1974).
- [24] P. F. Pérez, S. Ohmer and H. H. Patel, *Phys. Lett. B* **735**, 283 (2014).
- [25] P. F. Pérez and C. Murgui, *Phys. Lett. B* **777**, 381 (2018).
- [26] A. Celis, J. Fuentes-Martin, M. Jung and H. Serodio, *Phys. Rev. D* **92**, 015007 (2015).
- [27] R. Essig et al., FERMILAB-CONF-13-648, SLAC PUB-15960, 2003.
- [28] S. Alekhin, W. Altmannshofer, T. Asaka, B. Batell and F. Bezrukov et al., *Rep. Prog. Phys.* **79**, 124201 (2016).
- [29] T. Sjostrand, *PoS LHCP2016* **276**, 007 (2016).
- [30] EvtGen homepage, <https://evtgen.hepforge.org> (accessed Mar., 15, 2018).
- [31] PYTHIA homepage, <http://home.thep.lu.se/~torbjorn/Pythia.html> (accessed Mar., 15, 2018).
- [32] PHOTOS homepage, <http://photospp.web.cern.ch/photospp/> (accessed Mar., 15, 2018).
- [33] GEANT4 homepage, <http://geant4.web.cern.ch/geant4> (accessed Mar., 15, 2018).
- [34] S. Brandt, C. Peyrou, R. Sosnowski and A. Wroblewski, *Phys. Lett.* **12**, 57 (1964).
- [35] J. D. Bjorken and S.J. Brodsky, *Phys. Rev. D* **1**, 1416 (1970).
- [36] G. C. Fox and S. Wolfram, *Phys. Rev. Lett.* **41**, 1581 (1978).
- [37] R. E. Kalman, *J. Basic Eng.* **82**, 35 (1960).
- [38] C. M. Bishop, *Pattern Recognition and Machine Learning* (Springer, Singapore, 2006).
- [39] K. P. Murphy, *Machine Learning: A Probabilistic Perspective* (The MIT Press, 2012).
- [40] G. James, D. Witten, T. Hastie and R. Tibshirani, *An Introduction to Statistical Learning* (Springer, 2017).
- [41] T. Hastie, R. Tibshirani and J. Friedman, *The Elements of Statistical Learning* (Springer, 2016).
- [42] ROOT homepage, <https://root.cern.ch/> (accessed Mar., 15, 2018).
- [43] B. Efron and T. Hastie, *Computer Age Statistical Inference* (Cambridge University Press, 2016).
- [44] S. Amari, *Information Geometry And Its Applications* (Springer, Japan, 2016).