

Simulation von Teilchenwechselwirkungen für Praktika zur Detektorphysik

Simulation of particle interactions for courses
on detector physics

Bachelorarbeit
von

Maximilian Burkart

am Institut für Experimentelle Kernphysik

Referent: Prof. Dr. Günter Quast
Korreferent: Dr. Thorsten Chwalek

02. Februar 2017

Erklärung zur Selbstständigkeit

Ich versichere, dass ich diese Arbeit selbstständig verfasst habe und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe, die wörtlich oder inhaltlich übernommenen Stellen als solche kenntlich gemacht und die Satzung des KIT zur Sicherung guter wissenschaftlicher Praxis in der gültigen Fassung vom 27.11.2014 beachtet habe.

Karlsruhe, den 02. Februar 2017, _____
Maximilian Burkart

Als Ansichtsexemplar genehmigt von

Karlsruhe, den 02. Februar 2017, _____
Prof. Dr. Günter Quast

Inhaltsverzeichnis

1. Einleitung	1
2. Geant4	3
2.1. Struktur einer Simulation	3
2.2. Einbindung physikalischer Wechselwirkungen	4
2.3. Tracking	6
2.4. Detektordefinition	6
2.5. Geant4Py	8
3. Theoretische Grundlagen	9
3.1. Elektromagnetische Wechselwirkungen mit Materie	9
3.1.1. Energieverlust durch Ionisation	9
3.1.2. Wechselwirkungen von Elektronen	10
3.1.3. Wechselwirkungen von Photonen	10
3.1.4. Elektromagnetische Schauer	12
3.2. Hadronische Wechselwirkungen mit Materie	13
3.3. Kalorimeter	14
3.3.1. Elektromagnetische Kalorimeter	14
3.3.2. Hadronische Kalorimeter	15
4. Das Paket geant4_10_3_package	17
4.1. Konzeption	17
4.2. Aufbau	18
4.3. Visualisierung	19
4.4. Installation	20
4.5. Verwendung	20
5. Beispielsimulationen	23
5.1. Elektromagnetische Wechselwirkungen	23
5.2. Studien an einem elektromagnetischen Kalorimeter	26
5.3. Hadronisches Kalorimeter	28
5.4. Bestrahlung eines Tumors	31
6. Zusammenfassung und Ausblick	35
Literaturverzeichnis	37
Anhang	39
A. Übungsblatt	39

B. Dokumentation	39
----------------------------	----

Abbildungsverzeichnis

3.1. Energieverluste von Elektronen durch Ionisation und Bremsstrahlung.	11
4.1. Darstellung eines Ereignisses	21
5.1. Verteilung des Ortes der ersten Paarbildung	24
5.2. Länge und Breite eines elektromagnetischen Schauers.	26
5.3. Energieabhängigkeit der Länge und Breite elektromagnetischer Schauer.	27
5.4. Bestimmung der Kalibrationskonstante.	27
5.5. Energieabhängigkeit der Auflösung eines elektromagnetischen Kalorimeters.	28
5.6. Verteilung der Signale von Elektronen und Pionen.	30
5.7. Energieabhängigkeit des e/π -Verhältnisses.	31
5.8. Profil der Energiedeposition	32
5.9. Relative Energiedeposition im Tumor	33
B.1. Titelseite der Dokumentation	40

Tabellenverzeichnis

4.1. Farbgebung der Trajektorien der verschiedenen Teilchen.	20
5.1. Ergebnisse der Messung der Strahlungslänge.	25
5.2. Studie zur Länge des Kalorimeters	29

1. Einleitung

Das Ziel der vorliegenden Arbeit ist die Entwicklung einer *Python*-Anwendung des Toolkits Geant4 für die praktischen Übungen zu den Vorlesungen “Teilchenphysik I“ und “Detektoren für Teilchen- und Astroteilchenphysik“. Geant4 bietet einen objektorientierten, in der Programmiersprache C++ geschriebenen Ansatz der Simulation von Teilchenwechselwirkungen mit Materie.[Ago03]

Die Auswertung der Daten, die in modernen Experimenten der Hochenergiephysik produziert werden, erfordert eine Vielzahl verschiedener Softwarekomponenten.[Ago03] Zu diesen Komponenten gehören die Erzeugung der Ausgangszustände, die Simulation der Wechselwirkungen der Teilchen in Materie, die Berechnung der erwarteten Antwort des Detektors sowie Komponenten zur Analyse der Daten. Geant4 deckt den Bereich der Simulation der Wechselwirkungen und der Antwort des Detektors ab. Zu diesem Zweck stellt Geant4 Module zur Beschreibung der Geometrie eines Detektors, der verwendeten Materialien und Teilchen, zum Verfolgen der Teilchen durch den Detektor, zur Modellierung verschiedener physikalischer Wechselwirkungen, zur Visualisierung der simulierten Ereignisse und zum Speichern erzeugter Spuren zur Verfügung.[Ago03] Geant4 findet mittlerweile nicht nur Anwendung in der Experimenten der Hochenergiephysik, sondern wird auch in der Nuklearmedizin und der Strahlungs therapie sowie von ESA und NASA zur Simulation von Strahlungsschäden an Raumschiffen oder der Strahlungsdosis eines Astronauten verwendet.[All06]

Eine internationale Kollaboration bestehend aus mehr als 100 Wissenschaftlern und Ingenieuren zeigte sich für die Entwicklung des Toolkits verantwortlich.[Ago03] Seit der im Dezember 1998 veröffentlichten ersten Version[Ago03, All06] erfolgt eine stetige Weiterentwicklung der Funktionalität sowie eine Anpassung an neue technische Entwicklungen. Die aktuelle Version von Geant4 ist die am 9. Dezember 2016 veröffentlichte Version 10.3.

Die Beschäftigung mit Geant4 im Rahmen der praktischen Übungen dient dem Sammeln erster Erfahrungen mit der Simulation eines Detektors. Eine Schulung der Mechanismen, die im Detektor zur Generation des Signals dienen, ist ebenso sinnvoll, wie das Kennenlernen der Dimensionen eines Teilchenschauers. Die Analyse des Einflusses des Detektoraufbaus auf die Signalhöhen und die Auflösung fördert das Verständnis der grundlegenden Wechselwirkungen im Detektor. Dieses Wissen kann in der Konstruktion neuer Detektoren angewendet werden. Selbst für Studierende, die sich im weiteren Verlauf ihres Studiums auf die Datenanalyse spezialisieren, sind Studien der Detektorsimulation sinnvoll, da sie einen Einblick in die Generation der Monte-Carlo-Daten liefern, die zur Bestimmung der Selektionskriterien zur Analyse dienen.

Eine bereits vorhandene Anwendung benutzt eine Klasse des Datenanalyse-Frameworks *ROOT*[Bru97] zur Steuerung von Geant4. Das entwickelte Paket ersetzt diese Anwendung. Dies ist erforderlich, um eine direktere Verwendung von Geant4 während der Übungen zu erreichen. Das Fehlen einer ausreichenden Dokumentation der Schnittstellen ist ein

weiterer Aspekt, der für die Entwicklung einer neuen Anwendung spricht. In großen Softwareumgebungen kann die Syntax der Programmiersprache C++ von der Implementierung der Objekte ablenken, da zwei Dateien für jede Klasse angelegt werden. Dies führt zu einer Unübersichtlichkeit der Implementierung und erschwert dadurch die Einarbeitung. Die Wahl einer Programmiersprache für die Entwicklung fällt auf die Skriptsprache *Python*[Pyt]. Zum Einen existiert über die Bibliothek Geant4Py schon eine Schnittstelle des Toolkits zu *Python*, zum Anderen erleichtert die einfache Syntax von *Python* den Einstieg in die Anwendung und die Studierenden bringen einen großen Umfang an Vorwissen in *Python* mit, der im Bachelorstudium, sowie in den Anfängerpraktika vermittelt wird.

Das Ziel der Arbeit ist daher die Entwicklung einer gut strukturierten und dokumentierten *Python*-Anwendung, die die Steuerung vom Geant4 ermöglicht. Dabei gilt es einen Mittelweg zwischen der Abstraktion und Vereinfachung der Verwendung und der Nähe zur Struktur einer reinen Geant4-Anwendung zu finden.

Das erste Kapitel der Arbeit bildet eine Einführung in das Programmpaket Geant4. Ein besonderes Augenmerk liegt dabei auf der Beschreibung des Ablaufs einer Simulation. Daran schließt sich eine Erläuterung der theoretischen Grundlagen der Wechselwirkungen von Teilchen mit Materie an. Im dritten Kapitel erfolgt die Beschreibung des Aufbaus der entwickelten Anwendung in einem *Python*-Paket. Zusätzlich erläutert das Kapitel die Verwendung der Anwendung. Im fünften Kapitel erfolgt die Darstellung der Ergebnisse, die mit dem aktuellen Stand der Anwendung erhalten werden. Es handelt sich dabei im Wesentlichen um Simulationen der im dritten Kapitel vorgestellten theoretischen Grundlagen. Den Abschluss der Arbeit bildet eine kurze Zusammenfassung des Pakets sowie ein Ausblick auf zukünftige Entwicklungen. Im Anhang befindet sich die Dokumentation der zur Verfügung gestellten Schnittstellen.

2. Geant4

Dieses Kapitel dient der Einführung in die Grundlagen der Simulation eines Detektors mit Geant4. Der erste Abschnitt beschäftigt sich mit der Struktur und dem Ablauf einer Geant4-Anwendung. Der Fokus liegt dabei auf der Steuerung der Anwendung durch den Nutzer. Die beiden folgenden Abschnitte thematisieren die Funktionsweise der Teilchenverfolgung durch den Detektor und der Simulationsmechanismen. Der vierte Abschnitt des Kapitels beschäftigt sich mit der Definition einer Detektorgeometrie. Daran schließt sich die Beschreibung der Bibliothek Geant4Py an, die die Verwendung von Geant4 durch die Programmiersprache *Python*[Pyt] ermöglicht.

2.1. Struktur einer Simulation

Jede Geant4-Anwendung durchläuft während einer Simulation verschiedene Zustände. Diese sind der `preInit`-Zustand, ein Zustand während der Initialisierung, ein Zustand aus dem der Start eines Laufs erfolgt, in dem sich die Anwendung während des Laufs befindet und ein Stadium, das während des Verlassens der Anwendung durchlaufen wird. Der erste Schritt im Ablauf einer Simulation ist das Erzeugen einer Instanz der `RunManager`-Klasse, die den gesamten Ablauf kontrolliert.[Gea16a]

Das Erzeugen der `RunManager`-Instanz versetzt Geant4 in den `preInit`-Zustand. Aus diesem heraus erfolgt die Übergabe der Klassen, die der Beschreibung der Komponenten dienen, an den `RunManager`. [Gea16a] Es existieren drei erforderliche und fünf optionale Klassen. Die erforderlichen Klassen sind die `G4VUserDetectorConstruction`-Klasse, eine Physikliste und eine `G4PrimaryGeneratorAction`-Klasse, die der Erzeugung primärer Teilchen und Vertices dient. Die Definition der verwendeten Detektorgeometrie erfolgt durch die `G4VUserDetectorConstruction`-Klasse. Die `G4PrimaryGeneratorAction`-Klasse dient der Erzeugung des Anfangszustandes der Simulation. Der Anfangszustand kann zum Einen durch die Schnittstelle zu einem Framework zur Verfügung gestellt werden.[Ago03] Zum Anderen besteht durch die `G4ParticleGun`-Klasse die Möglichkeit, primäre Teilchen und Vertices zu erzeugen. Sie ermöglicht die Auswahl eines primären Teilchens und die Einstellung dynamischer Eigenschaften, wie Impuls, kinetische Energie, Ort und Flugrichtung. Außerdem besteht die Option, mehrere Teilchen auf einmal zu erzeugen oder dem Teilchen eine Polarisationsrichtung zuzuweisen.[Gea16a]

Nach der Übergabe der Klassen an den `RunManager` erfolgt die Initialisierung des Kernels. Sie beginnt mit dem Aufruf der `Initialize()`-Methode des `RunManagers`. Während der Initialisierung befindet sich die Anwendung im Initialisierungszustand und geht nach erfolgreicher Durchführung in den Bereitschaftszustand über.[Gea16a] Aus diesem heraus erfolgt der Start eines Simulationslaufs durch den Aufruf der `BeamOn()`-Methode der `RunManger`-Klasse. Als Argument erwartet sie die Anzahl an Ereignissen, die zu simulieren sind.

Die Simulation gliedert sich in verschiedene Simulationseinheiten, die hierarchisch aufgebaut sind. Die einzelnen Einheiten stellen immer kleinere Bausteine der Simulation dar.

Die größte Simulationseinheit ist ein Lauf. Ein Lauf besteht aus mehreren Ereignissen und wird durch den Aufruf der `BeamOn()`-Methode gestartet.[All16, Gea16a]

Ein Ereignis besteht aus dem Zerfall oder der Wechselwirkung des primären Teilchens oder der primären Teilchen.[All16] Zu Beginn der Simulation enthält das Ereignis Informationen über das primäre Teilchen und den primären Vertex. Diese werden während der Simulation umgewandelt und das Ereignis enthält nach der Simulation Informationen über die Trajektorien der Teilchen durch den Detektor sowie über im Detektor registrierte Treffer.[Ago03]

Die nächstkleinere Simulationseinheit ist eine Spur. Eine Spur repräsentiert ein Teilchen, das sich durch den Detektor bewegt.[Gea16a] Sie besteht aus mehreren Schritten. Eine Spur enthält statische Informationen über das transportierte Teilchen, wie die Ladung oder die Masse des Teilchens, genauso wie dynamische Eigenschaften, die sich während der Simulation ändern. Zu den dynamischen Eigenschaften gehören der Impuls, die kinetische Energie und der Ort, an dem sich das Teilchen befindet. Die Spur des Teilchens existiert bis das Teilchen zur Ruhe kommt oder zerfällt.[Gea16a]

Die Beschreibung des Ablaufs eines Schrittes erfolgt in Kapitel 2.3. Ein Schritt enthält Informationen über den Anfangs- und den Endpunkt.[Gea16a] Die Länge eines Schrittes ist beschränkt durch den Abstand zum nächsten Volumen, dem Energieverlust durch kontinuierliche Prozesse oder durch seine Beschränkung in der `G4UserLimit`-Klasse.[Ago03]

Die fünf zusätzlichen Klassen, mit denen der Kernel initialisiert werden kann, ermöglichen es, beim Übergang zwischen den Simulationseinheiten in die Verfolgung des Teilchens durch den Detektor einzugreifen.[Gea16a] Es existiert eine Klasse zum Eingriff in jede Simulationseinheit, sowie eine Klasse, mit der die Priorität der Verfolgung einer bestimmten Spur durch den Detektor verändert werden kann. Dies ist die `G4UserStackingAction`-Klasse. Mit den beiden Klassen `G4UserRunAction` und `G4UserEventAction` kann zu Beginn und am Ende eines Laufs, beziehungsweise eines Ereignisses in die Simulation eingegriffen werden. Verwendung finden diese Klassen in der Regel bei der Analyse eines Laufs oder Ereignisses.[Gea16a] Die Klasse `G4UserTrackingAction` dient dem Eingriff in das Verfolgen des Teilchens zu Beginn und am Ende einer Spur. Der Eingriff in den Ablauf eines Schrittes erfolgt durch die Klasse `G4UserSteppingAction`.

2.2. Einbindung physikalischer Wechselwirkungen

Die Integration physikalischer Wechselwirkungen in eine Simulation erfolgt über die Physikliste. Sie legt die Teilchen fest, die in der Simulation auftreten und welche Wechselwirkungen sie erfahren. Sie kann vom komplett vom Benutzer definiert werden. Zusätzlich besteht die Möglichkeit der Verwendung und Erweiterung einer vordefinierten Referenzphysikliste.[All16] Die Definition der Physikliste entspricht der Zuweisung aller Prozesse, die ein Teilchen erfahren kann, zu diesem Teilchen.[Ago03]

Die Repräsentation physikalischer Wechselwirkungen erfolgt durch die Geant4-Klasse `G4VProcess`. [Ago03] Der Begriff des Prozesses steht im Folgenden für die physikalischen Wechselwirkungen sowie die Beschreibung der Klasse `G4VProcess`. Die Schnittstellen aller Prozesse sind identisch. Dies ermöglicht eine allgemeine Behandlung aller Prozesse durch das Tracking. Die Abstraktion der Prozesse führt zu einer einfachen Möglichkeit, neue Prozesse zu einer Simulation hinzuzufügen oder bestehende Prozesse zu erweitern, um damit die Genauigkeit der Simulation zu verbessern.[Gea16a]

Prozesse gliedern sich in sieben verschiedene Kategorien. Diese sind elektromagnetische, hadronische, optische, Zerfalls- und photoleptonische Hadron-Prozesse. Zusätzlich existieren noch die beiden Kategorien der Transportprozesse und der Parametrisierung. Eine

weitere Unterteilung der Prozesse erfolgt nach dem Typ der Wechselwirkung. Unterschieden werden Prozesse für Teilchen, die sich in Ruhe befinden, die entlang des gesamten Schrittes stattfinden, und Prozesse, die lokal am Ende des Schritts auftreten. Ergänzt wird diese Unterscheidung durch Prozesse, die sich aus mehreren verschiedenen Prozesstypen zusammensetzen.[Ago03]

Jeder Prozess besitzt die Methoden `GetPhysicalInteractionLength()` und `DoIt()`. [Ago03] Die `DoIt()`-Methode dient der Durchführung des Prozesses. Die `GetPhysicalInteractionLength()`-Methode berechnet die Schrittlänge, die zur Verfolgung des Teilchens durch den Detektor verwendet wird. Die Schrittlänge ist der Abstand zwischen dem aktuellen Ort des Teilchens und dem Ort der Wechselwirkung.[Ago03] Ihre Bestimmung erfolgt über die mittlere freie Weglänge.[Ago03] Die mittlere freie Weglänge gibt die Länge an, nach der eine Wechselwirkung im Mittel auftritt. Sie ist abhängig von der Energie des betrachteten Teilchens und durch

$$\lambda(E) = \left(\sum_i n_i \cdot \sigma(E, Z_i) \right)^{-1} \quad (2.1)$$

definiert.[Gea16b] Hierbei bezeichnet σ den totalen Wirkungsquerschnitt der Wechselwirkung, der abhängig von der Energie E des Teilchens und der Kernladungszahl der Materialien Z_i ist. Die Summation läuft über alle Komponenten des Materials, in dem sich das Teilchen gerade befindet. n_i gibt die Atomdichte der i -ten Komponente an, die sich über

$$n_i = \frac{N_A \rho w_i}{A_i} \quad (2.2)$$

berechnet. N_A steht hierbei für die Avogadro-Konstante, ρ für die Dichte des Materials, w_i für den Massenanteil und A_i für die molare Masse der i -ten Komponente.[Gea16b]

Die Materialabhängigkeit der mittleren freien Weglänge verhindert die direkte Verwendung zur Bestimmung der Schrittlänge.[Gea16b] Stattdessen verwendet Geant4 die Wahrscheinlichkeit der Durchquerung einer Länge l ohne Wechselwirkung, die durch

$$P(l) = e^{-n_\lambda l} \quad (2.3)$$

gegeben ist.[Ago03] n_λ ist die Anzahl der mittleren freien Weglängen, die das Teilchen zurücklegt, ohne zu wechselwirken. Sie berechnet sich über

$$n_\lambda = \int_0^l \frac{dx}{\lambda(x)}. \quad (2.4)$$

n_λ ist energie- und materialunabhängig und kann deshalb verwendet werden, um die Schrittlänge zu bestimmen. Die Berechnung der Anzahl an Schrittlängen erfolgt durch Transformation der zwischen 0 und 1 gleichverteilten Zufallszahl η . [Ago03, Gea16b] Damit berechnet sich n_λ über

$$n_\lambda = -\ln(\eta). \quad (2.5)$$

Die Länge eines Schrittes berechnet sich durch

$$s(x) = n_\lambda \cdot \lambda(x). \quad (2.6)$$

2.3. Tracking

Die Abstraktion der physikalischen Wechselwirkungen in Prozesse mit identischen Schnittstellen ermöglicht es, den Transport eines beliebigen Teilchens durch den Detektor mit einem Algorithmus zu beschreiben. Die Verfolgung eines Teilchens durch den Detektor gestaltet sich somit unabhängig von den betrachteten Teilchen und physikalischen Wechselwirkungen. In Geant4 erfolgt der Transport der Teilchen durch den Detektor schrittweise.[Ago03]

Zu Beginn des Schritts schlägt jeder Prozess aus der Liste des betrachteten Teilchens über seine `GetPhysicalInteractionLength()`-Methode eine Schrittlänge vor.[Ago03] Befindet sich das Teilchen in Ruhe, so werden nur Zerfallsprozesse betrachtet. Für Teilchen im Flug konkurrieren alle Prozesstypen. Um die Genauigkeit der Simulation zu verbessern, existieren mehrere Mechanismen, die die Schrittlänge eines Teilchens zusätzlich begrenzen. Zum Einen schlagen auch Prozesse, die einen kontinuierlichen Energieverlust beschreiben, eine Schrittlänge vor. Dies ist erforderlich, um die Änderung des Wirkungsquerschnittes durch den Energieverlust während eines Schritts gering zu halten.[Gea16b] Weiterhin begrenzt der kürzeste Abstand des aktuellen Ortes zur nächsten Volumengrenze die Schrittlänge. Dies gewährleistet, dass das Teilchen während des Schritts in kein anderes Volumen übertritt.[All16]

Die kleinste vorgeschlagene Schrittlänge bestimmt den Prozess, der ausgeführt wird. Prozesse, die mit einem Energieverlust oder einer Richtungsänderung des Teilchens verbunden sind, können ihre Ausführung erzwingen und finden auch statt, falls ihre vorgeschlagene Schrittlänge nicht die kürzeste ist.[Ago03] Die `DoIt()`-Methode dieser Prozesse wird als Erstes aufgerufen. Zur Ausführung eines Prozesses gehört die Berechnung der Energieänderung des Teilchens, die sich aus der Summe der Energieänderungen aller Prozesse zusammensetzt, sowie das Erzeugen sekundärer Teilchen.[Gea16a] Nach der Ausführung aller Prozesse erfolgt eine Aktualisierung der Eigenschaften des Teilchens. Zu diesen zählen der Ort, an dem sich das Teilchen befindet, seine kinetische Energie und die aktuelle Zeit. Nach der Ausführung der kontinuierlichen Prozesse findet die Durchführung der diskreten Prozesse statt.[Gea16a] Anschließend erfolgt die Aktualisierung der Eigenschaften des Teilchens und des Abstandes zur nächsten Volumengrenze. Stammt die kürzeste Schrittlänge aus dem Abstand zum nächsten Volumen, so finden keine diskreten Prozesse statt und das Teilchen wandert an den Beginn des nächsten Volumens.[Gea16a]

Den Schluss eines Schritts bildet die Aktualisierung der Anzahl an mittleren freien Weglängen, die das Teilchen passiert, bevor es wechselwirkt, für jeden Prozess, der nicht stattgefunden hat, nach der Formel

$$n'_\lambda = n_\lambda - \frac{\Delta x}{\lambda(x)}. \quad [\text{Gea16b}] \quad (2.7)$$

Δx bezeichnet hierbei die Strecke, die das Teilchen in diesem Schritt zurückgelegt hat. Für den aufgetretenen Prozess, wird ein neues n_λ erzeugt und der nächste Schritt beginnt.[Gea16a]

2.4. Detektordefinition

Ein Objekt der `G4VUserDetectorConstruction`-Klasse beschreibt den Aufbau einer Detektorgeometrie. Hierzu gehört die Definition der geometrischen Zusammensetzung, die Beschreibung der verwendeten Materialien, die Definition der elektromagnetischen Felder, die in den einzelnen Detektorregionen vorliegen, sowie der elektronischen Eigenschaften der einzelnen Komponenten. Zusätzlich besteht die Möglichkeit, den einzelnen Volumen Visualisierungsattribute und benutzerdefinierte Beschränkungen zuzuweisen.[Gea16a]

Geometrie

Die Anforderungen an die Geometriedefinition sind vielfältig. Sie reichen von grundlegenden Analysen zu Kalorimetern bis zu komplexen Detektoraufbauten an Großexperimenten wie zum Beispiel am Large Hadron Collider.[Apo09] Die Definition der geometrischen Objekte, die ein Detektor enthält, erfolgt in Geant4 in drei Stufen.

Die erste Stufe ist die Definition eines Körpers. Ein Körper ist definiert durch seine Form und seine Dimensionen. Beispiele für einen Körper sind eine Vollkugel mit einem Radius von 4,5 cm oder ein Quader mit einer Seitenlänge von 20 cm.[Gea16a] Die Konstruktion der Körper erfolgt durch Auswählen der passenden Form aus den zur Verfügung gestellten *Constructed Solid Geometries* (CSG).[Apo09]

Die zweite Stufe der Geometriedefinition erfolgt durch Hinzufügen physikalischer Eigenschaften zu bereits definierten Volumina. Das entstehende Objekt heißt logisches Volumen und wird durch die Klasse `G4LogicalVolume` repräsentiert. Das logische Volumen erhält seine physikalischen Eigenschaften durch das Material, aus dem es besteht. Auch die Definition der elektromagnetischen Felder und der benutzerdefinierten Beschränkungen gehört zu einem logischen Volumen.[Gea16a]

Die dritte und letzte Stufe der Definition eines Detektors ist die Positionierung des logischen Volumens im Raum. Ein platziertes Volumen heißt physikalisches Volumen. Um einen Detektor vollständig zu beschreiben, ist es notwendig, dass Volumen ineinander platziert werden können. Das Welt-Volumen stellt das größte Volumen in der Definition eines Detektors dar. Es enthält alle anderen Volumina, die den Detektor beschreiben. Das platzierte Volumen wird Tochtervolumen genannt und ist vom Muttervolumen umschlossen. Die Position des Tochtervolumens wird relativ zum Mittelpunkt des Muttervolumens angegeben.[Gea16a]

Materialien

Der Aufbau der Materialien in Geant4 spiegelt den Aufbau von Materialien in der Natur wieder. Materialien sind aus Molekülen oder Elementen aufgebaut, die sich wiederum aus Isotopen zusammensetzen.[Ago03] Die definierenden Eigenschaften eines Isotops sind der Name des Isotops, die Kernladungszahl, die Nukleonenzahl sowie die molare Masse. Ein Element besitzt die Eigenschaft des Namens, der Kernladung, der effektiven Nukleonenzahl, der effektiven molaren Masse und des Wirkungsquerschnitts pro Atom.[Gea16a]

Der Zugriff auf ein Element erfolgt über dessen Symbol im Periodensystem der Elemente. Die Definition eines Elements erfolgt entweder durch die Zusammensetzung der Isotope oder direkt durch Definition der effektiven Größen. Der Wirkungsquerschnitt pro Atom berechnet sich aus der Kernladung, der Nukleonenzahl und der molaren Masse.[Gea16a] Analog zur Definition eines Elements aus Isotopen erfolgt die Definition eines Materials. Entweder wird ein neues Element mit den effektiven Werten erzeugt oder verschiedene Elemente werden zu einem Material zusammengefügt. Ein Material wird durch seine Eigenschaften wie Dichte, Aggregatzustand, Temperatur und Druck definiert. Aus diesen Größen sowie den Eigenschaften der zugrundeliegenden Elemente berechnet Geant4 die für die Simulation verwendeten Größen mittlere freie Weglänge, Strahlungslänge, hadronische Wechselwirkungslänge und den mittleren Energieverlust pro Strahlungslänge, der durch die Bethe-Gleichung (siehe Gleichung 3.1) gegeben ist.[Gea16a]

Die Definition von Materialien erfolgt direkt durch den Benutzer des Toolkits. Die Werte der physikalischen Größen müssen hierbei im Programm-Code definiert werden. Weiterhin besteht die Möglichkeit, Materialien aus der internen Datenbank zu definieren. Dies erleichtert die Definition der Materialien, da alle physikalischen Größen eines Isotops, Elementes oder Materials zur Verfügung gestellt werden. In der Datenbank befinden

sich etwa 3500 Isotope, 108 Elemente und etwa 300 Materialien.[Apo09] Die Daten der Einträge stammen aus der Datenbank des National Institute of Standards and Technology (NIST).[Cou15]

G4UserLimits

Um die Laufzeit einer Simulation zu verringern, verhindert Geant4 die Produktion niederenergetischer Teilchen. Die Energie, die für die Erzeugung des Teilchens verwendet worden wäre, wird im Detektor deponiert.[All16] Die Produktionsschwelle für die Teilchen ist durch ihre Reichweite im Material gegeben und liegt in den Referenzphysiklisten bei 0,7 mm.[All16] Aus der Schwelle auf die Reichweite berechnet Geant4 die Energie, ab der die Teilchen nicht mehr weiter verfolgt werden. Zur Optimierung einer Anwendung bietet es sich an, die Schnitte auf die Reichweite zu modifizieren. Dies geschieht entweder durch Anpassung der Schnitte in gewünschten Detektorregionen oder durch die Klasse `G4UserLimits`. [Gea16a] Sie bietet die Möglichkeit der Beschränkung der Schrittlänge, der maximalen Länge einer Spur, der maximalen Zeit der Verfolgung, der Mindestenergie, die ein Teilchen besitzen darf, und der Reichweite. Die Integration der Schnitte in eine Simulation erfolgt durch die Zuweisung einer Instanz der `G4UserLimits`-Klasse zu einem logischen Volumen.

2.5. Geant4Py

Die Programmbibliothek `Geant4Py` bietet die Möglichkeit, die C++-Klassen von Geant4 direkt über die Skriptsprache *Python*[Pyt] zu verwenden. Dies wird erreicht, indem grundlegende Geant4-Klassen mit Hilfe der Boost-Bibliotheken verpackt und so die Schnittstellen für *Python* zugänglich gemacht werden.[Gea16a] Die Einbindung der verpackten Klassen in den *Python*-Code funktioniert durch den Import des zur Verfügung gestellten *Python*-Moduls "Geant4". Zusätzlich zum Verpacken der Klassen werden auch Instanzen aller Manager-Klassen sowie weiterer funktioneller Elemente von Geant4 erzeugt. Der Zugriff auf diese Instanzen erfolgt durch Voranstellen eines Buchstabens `g` vor dem Namen der Klasse, `gRunManager` zum Beispiel ist eine Instanz der `RunManager`-Klasse. Außerdem werden noch drei Nebenmodule zur Verfügung gestellt.[Gea16a] Diese dienen der vereinfachten Geometrie- und Materialdefinition sowie der vereinfachten Verwendung der `G4ParticleGun`-Klasse.

3. Theoretische Grundlagen

Dieses Kapitel beschäftigt sich mit den Wechselwirkungen von Teilchen mit Materie. Sie bilden die Grundlagen, der in Kapitel 5 vorgestellten Simulationen. Zuerst erfolgt die Diskussion elektromagnetischer Wechselwirkungen. Daran schließen sich die hadronischen Wechselwirkungen an und den Abschluss des Kapitels bildet die Anwendung der Wechselwirkungen in Kalorimetern zur Energiemessung von Teilchen.

3.1. Elektromagnetische Wechselwirkungen mit Materie

3.1.1. Energieverlust durch Ionisation

Geladene Teilchen, die sich durch ein Medium bewegen, verlieren Energie durch einzelne, stochastisch auftretende Stöße mit den Atomen des Materials.[Bic06] Durch den Stoß erfolgt eine Ionisation, Anregung des Atoms oder eine kollektive Anregung des Mediums.[Pat16] Der Energieverlust in einer Kollision ist meist gering. In seltenen Fällen wird ein größerer Teil der Energie des Teilchens übertragen.[Pat16]

Die Bethe-Formel gibt den mittleren Energieverlust schwerer geladener Teilchen an. In der Form, wie sie in [Pat16] angegeben ist, lautet die Bethe-Formel

$$\left\langle -\frac{dE}{dx} \right\rangle = K z^2 \frac{Z}{A} \frac{1}{\beta^2} \left[\frac{1}{2} \ln \frac{2m_e c^2 \beta^2 \gamma^2 W_{\max}}{I^2} - \beta^2 - \frac{\delta(\beta\gamma)}{2} \right]. \quad (3.1)$$

W_{\max} bezeichnet den maximalen Energieübertrag während einer Kollision. Er ist durch

$$W_{\max} = \frac{2m_e c^2 \beta^2 \gamma^2}{1 + 2\gamma \frac{m_e}{M} + \left(\frac{m_e}{M}\right)^2} \quad (3.2)$$

gegeben.[Pat16] Der Vorfaktor K berechnet sich über $4\pi N_A r_e^2 m_e c^2$. z bezeichnet die Ladung des Teilchens, Z die Ladungszahl und A die Massenzahl des Materials, N_A die Avogadro-Konstante, m_e die Elektronenmasse, r_e den Elektronenradius, c die Lichtgeschwindigkeit, I die mittlere Anregungsenergie des Materials, δ eine Dichtekorrektur, M die Masse des Teilchens und die beiden Parameter β und γ die relativistischen, kinematischen Größen. Die Bethe-Formel ist im Bereich $0.1 \lesssim \beta\gamma \lesssim 1000$ gültig.[Pat16]

Für kleine Teilchenenergien dominiert der $1/\beta^2$ -Term in der Bethe-Formel. Dies führt dazu, dass Teilchen, die ihre Energie nur durch Ionisationsprozesse im Material deponieren, eine feste Reichweite besitzen und ihre Energiedeposition beim Erreichen dieser Reichweite am größten ist.[Kol16] Der charakteristische Peak in der Verteilung der Energiedeposition wird Bragg-Peak genannt.

3.1.2. Wechselwirkungen von Elektronen

Beim Durchqueren eines Materials können Elektronen ihre Energie auf zwei verschiedene Arten deponieren.[Kol16] Zum Einen deponieren Elektronen ihre Energie durch die Ionisation des Mediums, zum Anderen verlieren sie ihre Energie durch Erzeugung von Bremsstrahlung.

Der Energieverlust von Elektronen durch Ionisation weicht in geringem Maße vom Ionisationsverlust schwerer geladener Teilchen ab. Der Grund für diese Abweichung sind die Kinematik, der Spin, die Ladung und die Tatsache, dass es sich bei der in der Ionisation betrachteten Streuung um die Streuung zweier identischer Teilchen handelt.[Pat16]

Bei der Bremsstrahlung handelt es sich um den Energieverlust geladener Teilchen im Coulomb-Feld eines Atomkernes durch die Abstrahlung eines Photons. Die Bremsstrahlung verläuft somit analog zu einer Rutherfordstreuung unter Abstrahlung eines Photons.[Kol16] Der mittlere Energieverlust durch Bremsstrahlung kann näherungsweise durch

$$\left(\frac{dE}{dx}\right)_{\text{rad}} \simeq -\frac{E}{X_0} \quad (3.3)$$

über die Strahlungslänge X_0 ausgedrückt werden.[Wig00] Hierbei ist die Strahlungslänge durch

$$X_0 = \frac{716.405A}{Z^2(L_{\text{rad}} - f) + ZL'_{\text{rad}}} \quad (3.4)$$

gegeben.[Tsa74] A bezeichnet die molare Masse des Materials, f die Coulombkorrektur und L_{rad} und L'_{rad} den elastischen und inelastischen Streulogarithmus. Durch Integration des mittleren Energieverlusts ergibt sich

$$E(x) = E_0 e^{-\frac{x}{X_0}} \quad (3.5)$$

und somit ein exponentieller Abfall der Energie des Elektrons mit wachsender Eindringtiefe.

Da die Energieverluste durch die Ionisation logarithmisch mit der Energie wachsen, während die Bremsstrahlungsverluste linear mit der Energie zunehmen[Pat16], dominieren die Bremsstrahlungsverluste für hohe Energien über die Verluste durch Ionisation. Mit sinkender Elektronenenergie beginnen die Verluste durch Ionisation zu dominieren. Dies ist in Abbildung 3.1 dargestellt.

Die kritische Energie ϵ spielt eine wichtige Rolle in der Beschreibung der Entwicklung elektromagnetischer Schauer. Für sie existieren zwei verschiedene Definition. In der Ersten entspricht die kritische Energie der Energie, bei der die Strahlungsverluste durch die Ionisation gleich groß sind wie die Bremsstrahlungsverluste.[Fab03] In der Definition nach Rossi[Ros52] entspricht die kritische Energie der Energie, bei der die Energieverluste durch Ionisation pro Strahlungslänge der Energie des Elektrons entsprechen. Die Definition der kritischen Energie nach Rossi liefert bessere Resultate bei der Beschreibung der lateralen Ausdehnung elektromagnetischer Schauer.[Pat16] In der in Gleichung 3.3 angegebenen Näherung stimmen beide Definitionen der kritischen Energie überein.[Pat16]

3.1.3. Wechselwirkungen von Photonen

Photonen wechselwirken hauptsächlich auf vier verschiedene Arten mit Materie.[Wig00] Die möglichen Prozesse sind der Photoeffekt, die Rayleigh-Streuung, die Compton-Effekt und die Paarbildung.

Der Photoeffekt stellt das Herausschlagen eines Elektrons aus der Hülle eines Atoms dar.[Wig00] Der frei gewordene Platz in der Schale wird entweder durch ein Augerelektron

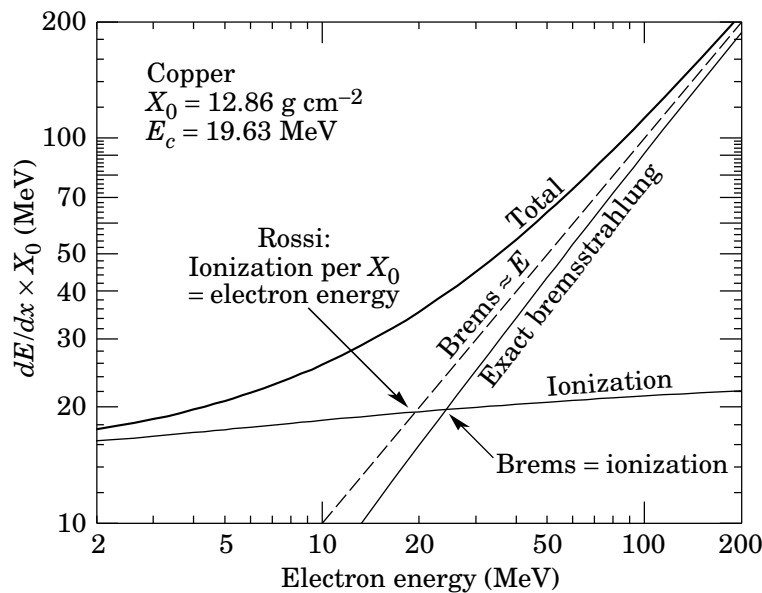


Abbildung 3.1.: Energieverluste von Elektronen durch Ionisation und Bremsstrahlung. Dargestellt ist das Verhalten der Energieverluste für verschiedene Energien in Kupfer. Die beiden Definitionen der kritischen Energie sind eingezeichnet und mit Pfeilen markiert. Entnommen aus [Oli14]

oder durch Abstrahlung charakteristischer Röntgenstrahlung wieder aufgefüllt. Da der Wirkungsquerschnitt für den Photoeffekt proportional zu E^{-3} ist [Wig00], ist der Photoeffekt dominant für niedrige Photonenenergien.

Bei der Rayleigh-Streuung handelt es sich um die kohärente elastische Streuung von Photonen an den Hüllenelektronen eines Atoms. Bei diesem Prozess findet keine Energiedeposition im Material statt, das Photon wird lediglich abgelenkt. [Wig00]

Der Compton-Effekt beschreibt die Streuung des einfallenden Photons an einem freien oder quasi-freien Elektron aus einer äußeren Schale der Hülle eines Atoms. [Wig00] Die Bezeichnung quasi-frei bedeutet, dass die Energie des Photons deutlich größer als die Bindungsenergie des Elektrons ist. [Kol16] Die Dominanz des Compton-Effekts gegenüber anderen Prozessen erstreckt sich über einen Energiebereich von einigen hundert keV bis ungefähr 5 MeV. [Wig00] Der Wirkungsquerschnitt für den Compton-Effekt ist proportional zu $1/E$. [Wig00]

Ist die Energie des einfallenden Photons größer als die doppelte Elektronenmasse, besteht die Möglichkeit, dass das Photon im Coulomb-Feld eines Kerns ein Elektron-Positron-Paar erzeugt. Der Prozess der Paarbildung ist im Vakuum nicht möglich, da aus Gründen der Impulserhaltung ein Partner benötigt wird, der den Rückstoß aufnimmt. Dieser Partner ist in 99 % der Fällen ein Atomkern. [Wig00]

Da das Photon bei der Erzeugung des Elektron-Positron-Paares vernichtet wird, handelt es sich bei der Paarbildung wie beim Photoeffekt um einen Absorptionsprozess. Charakteristisch für einen Absorptionsprozess ist der exponentielle Abfall der Intensität eines einfallenden Teilchenstrahles mit der Eindringtiefe. Die Intensität des Strahles ist durch

$$I(x) = I_0 e^{-\frac{7}{9} \frac{x}{X_0}} \quad (3.6)$$

gegeben. [Fab03]

Die Möglichkeit der Verwendung der Strahlungslänge als charakteristische Länge der Paarbildung ergibt sich aus der Ähnlichkeit der Wirkungsquerschnitte von Bremsstrahlung

und Paarbildung. Dies liegt an der Ähnlichkeit der Feynman-Diagramme der beiden Prozesse.[Pat16] Der Wirkungsquerschnitt für die Paarbildung wächst mit der Energie und erreicht für hohe Energien ein Plateau.[Wig00]

3.1.4. Elektromagnetische Schauer

Elektromagnetische Schauer entstehen durch den Einfall hochenergetischer Photonen, Positronen oder Elektronen. Die Entwicklung eines Schauers erfolgt über die Prozesse der Paarbildung und Bremsstrahlung.[Wig00]

Durch den Einfall eines hochenergetischen Photons bildet sich ein Elektron-Positron-Paar. Ist die Energie der beiden Teilchen hoch genug, erzeugen sie durch Bremsstrahlung weitere Photonen, die, falls sie genügend Energie erhalten, zu einer erneuten Paarbildung führen. Ein einfallendes Elektron erzeugt zunächst Photonen, die zu Paarbildungen führen können. Die meisten der durch Bremsstrahlung erzeugten Photonen besitzen nicht genügend Energie, um eine Paarbildung anzuregen.[Wig00] Diese Photonen verlieren zunächst einen Großteil ihrer Energie durch mehrfache Compton-Streuung und werden schließlich über einen Photoeffekt absorbiert. Die in diesen Prozessen erzeugten Elektronen deponieren ihre Energie durch Ionisation. Mindestens die Hälfte der Energiedeposition in einem Schauer erfolgt über diesen Mechanismus.[Wig00]

Zu Beginn der Entwicklung eines Schauers entstehen viele neue Teilchen. Durch diese Erhöhung der Teilchenmultiplizität verteilt sich die Energie des einfallenden Teilchens auf eine große Anzahl weiterer Teilchen und die Energiedeposition im Schauer wächst.[Wig00] Der Punkt, an dem die Teilchenmultiplizität nicht mehr weiter steigt, wird als Schauermaximum definiert. Nach Erreichen des Schauermaximums ist die Energie der Elektronen und Positronen unter die kritische Energie gefallen. Die Teilchen deponieren ihre Energie nur noch über die Ionisation und erzeugen somit keine Photonen mehr. Mit der dadurch bedingten Abnahme der Teilchenmultiplizität nimmt auch die Energiedeposition ab. Fabjan und Gianotti [Fab03] geben die Lage des Schauermaximums in Einheiten der Strahlungslänge durch

$$t_{\max} = \ln \left(\frac{E}{\epsilon} \right) + t_0 \quad (3.7)$$

an. E bezeichnet hierbei die Energie des einfallenden Teilchens und ϵ die kritische Energie. Der Parameter t_0 ist unterschiedlich für Elektronen und Photonen und beträgt für Elektronen $-0,5$ und für Photonen $0,5$. Somit wächst die Länge eines elektromagnetischen Schauers logarithmisch mit der Energie.

Die transversale Ausdehnung eines elektromagnetischen Schauers entsteht durch die Vielfachstreuung von Elektronen und Positronen aus der Strahlrichtung. Zusätzliche Beiträge liefern Photonen, die durch Bremsstrahlung von Elektronen, die sich von der Strahlachse entfernen, entstehen. Auch isotrope Streuprozesse von Photonen tragen zur Verbreiterung des Schauers bei.[Wig00, Fab03] Das transversale Profil eines elektromagnetischen Schauers ist daher energieunabhängig und durch den Molière-Radius charakterisiert.[Fab03] Dieser ist durch

$$R_M = 21 \text{ MeV} \frac{X_0}{\epsilon (\text{MeV})} \quad (3.8)$$

gegeben. ϵ steht in diesem Fall für die kritische Energie in der Definition nach Rossi. Innerhalb des Molière-Radius findet 90 % der Energiedeposition im Schauer statt.[Fab03, Pat16]

3.2. Hadronische Wechselwirkungen mit Materie

Die Entwicklung hadronischer Schauer ist deutlich komplizierter als die elektromagnetischer Schauer.[Wig00] Dies liegt daran, dass in elektromagnetischen Schauern nur wenige Prozesse eine Rolle spielen. Aufgrund der vielfältigeren starken Wechselwirkung treten bei der Entwicklung hadronischer Schauer mehr verschiedene Prozesse auf. Ein weiterer Aspekt, der zur Komplexität hadronischer Schauer beiträgt, ist, dass ein getroffener Kern nukleare Wechselwirkungen erfährt. In elektromagnetischen Schauern hingegen dient das Target lediglich zum Streuen der am Schauer beteiligten Teilchen.[Wig00]

Geladene Hadronen deponieren einen Teil ihrer Energie über die Ionisation des Mediums, bis sie in einem Absorptionsprozess hochenergetische sekundäre Teilchen produzieren. Im Gegensatz dazu deponieren neutrale Hadronen ihre Energie nur durch Absorption.[Wig00, Fab03] Die mittlere freie Weglänge zwischen zwei hadronischen Wechselwirkungen ist durch die hadronische Absorptionslänge $\lambda_{\text{had}} = 35 A^{1/3} \text{ g cm}^{-2}$ gegeben.[Fab03] A bezeichnet die Massenzahl des durchquerten Materials. Die im Absorptionsprozess erzeugten Hadronen propagieren weiter durch den Detektor, bis sie selbst absorbiert werden.

Die Produktion sekundärer Hadronen im Kern findet über den Prozess der Spallation statt. Die Spallation gliedert sich in zwei Phasen,[Wig00] die intranukleare Kaskade und die Evaporation.

Bei der intranuklearen Kaskade streut das einfallende Hadron an quasi-freien Nukleonen im Kern. Diese Nukleonen propagieren weiter durch den Kern und streuen an anderen Nukleonen. Es bildet sich eine Kaskade an Teilchen im Kern aus. Während der Ausbildung der intranuklearen Kaskade bilden sich Pionen und andere instabile Hadronen. Einige der so erzeugten Teilchen entkommen dem Kern und propagieren weiter durch das Medium und tragen so zur Entwicklung des hadronischen Schauers bei. Die Energien der Teilchen, die weiter durch das Medium propagieren sind im GeV-Bereich.[Fab03] Teilchen, die nicht aus dem Nukleus entkommen, führen zu einer Anregung des Kerns. Durch Abstrahlung freier Nukleonen, α -Teilchen oder schwererer Teilchen verliert der Kern diese Anregungsenergie wieder. Die restliche im Kern verbliebene Energie wird über Photonen abgestrahlt. Die Energie, die in diesen beiden Prozessen vom Kern abgestrahlt wird, ist in der Größenordnung einiger MeV.[Fab03]

Die Teilchen, die zur Entwicklung der hadronischen Kaskade führen, sind Protonen, Neutronen sowie geladene und neutrale Mesonen.[Fab03] Die meisten der produzierten Mesonen sind Pionen. Ein Drittel aller produzierten Pionen sind neutrale Pionen, die elektromagnetisch in zwei Photonen zerfallen. Dieser Zerfall tritt auf, bevor die neutralen Pionen hadronisch wechselwirken können und führt dazu, dass ein Anteil der Energie des hadronischen Schauers in einen elektromagnetischen Subschauer überführt wird.[Fab03] Da der übertragene Energieanteil nicht mehr für hadronische Wechselwirkungen zur Verfügung steht, wächst der Anteil des elektromagnetischen Subschauers mit der Energie des einfallenden Hadrons. Nach Gabriel [Gab94] beschreibt das Potenzgesetz

$$f_{\text{em}} = 1 - \left(\frac{E}{E_0} \right)^k \quad (3.9)$$

den elektromagnetischen Anteil des hadronischen Schauers. Hierbei gibt E die Energie des einfallenden Hadrons an. E_0 ist die Energie, die benötigt wird, um neutrale Pionen zu erzeugen und der Exponent k ist durch

$$k = \frac{\ln(1 - f_{\pi^0})}{\ln n} \quad (3.10)$$

gegeben. f_{π^0} gibt den relativen Anteil neutraler Pionen an, die in einer Kollision erzeugt werden, und n ist die mittlere Anzahl an produzierten Hadronen in der ersten Kollision. Für

den relativen Anteil der produzierten neutralen Pionen wird der Wert $1/4$ verwendet. Für $n = 5$ bis $n = 10$ befindet sich der Wert für k zwischen $k = -0,18$ und $k = -0,13$. [Gab94] Der elektromagnetische Anteil eines einzelnen Schauers fluktuiert stark um den mittleren Wert f_{em} , da der elektromagnetische Anteil von den Prozessen abhängt, die zu Beginn des Schauers stattfinden. [Wig00]

Im Gegensatz zu elektromagnetischen Schauern, die ihre gesamte Energie über die Ionisation deponieren, ist die Energie in einem hadronischen Schauer nicht vollständig detektierbar. [Wig00] Der Grund dafür ist, dass verzögerte Photonen, weiche Neutronen und die Bindungsenergie der Hadronen und Nukleonen unsichtbar für die Energiemessung sind. [Fab03]

Durch Unterschiede in den Wirkungsquerschnitten der elektromagnetischen und der starken Wechselwirkung besitzen hadronische Schauer eine deutlich größere räumliche Ausdehnung als elektromagnetische. [Wig00]

3.3. Kalorimeter

Kalorimeter dienen der destruktiven Energiemessung durch Aufschauern einfallender Teilchen. Abhängig von der Identität der vermessenen Teilchen erfolgt eine Unterteilung in elektromagnetische und hadronische Kalorimeter.

Kalorimeter teilen sich in homogene und Sampling-Kalorimeter ein. Homogene Kalorimeter bestehen aus einem Material, das sowohl als Absorber für die Teilchen fungiert als auch das Signal generiert, das gemessen werden kann. Sie bestehen aus inorganischen, schweren Szintillationskristallen oder nicht szintillierenden Cherenkov-Strahlern. [Pat16] Sampling-Kalorimeter bestehen aus einer Abfolge aus aktiven und passiven Schichten. In den passiven Schichten werden die Teilchen absorbiert und in den aktiven Schichten erfolgt die Erzeugung des Signals durch Ionisation oder Szintillation. Materialien, die in passiven Schichten Verwendung finden, sind Blei, Eisen, Kupfer und Uran. In aktiven Schichten werden flüssige Edelgase, organische oder anorganische Szintillatoren verwendet. [Pat16]

Die beiden nachfolgenden Unterkapitel beschäftigen sich mit den Eigenschaften elektromagnetischer und hadronischer Sampling-Kalorimeter.

3.3.1. Elektromagnetische Kalorimeter

Die relative Energieauflösung elektromagnetischer Kalorimeter ist durch

$$\frac{\sigma}{E} = \frac{a}{\sqrt{E}} \oplus b \oplus \frac{c}{E} \quad (3.11)$$

gegeben. [Fab03, Pat16] Das Symbol \oplus steht für die quadratische Summe der einzelnen Terme. Der erste Term ist der stochastische Term, der zweite der konstante und der dritte der Rauschterm.

Der stochastische Term kommt durch Fluktuationen der Anzahl der geladenen Spuren im aktiven Medium zustande. Nach Amaldi [Ama81] ist der stochastische Term in Sampling-Kalorimetern proportional zu

$$\frac{\sigma}{E} \propto \sqrt{\frac{t}{E}}. \quad (3.12)$$

Hierbei beschreibt t die Dicke des Absorbers in Einheiten der Strahlungslänge X_0 und E ist die Energie des einfallenden Teilchens. Um diese Proportionalität zu erhalten ist die Annahme notwendig, dass die Anzahlen der geladenen Spuren in den einzelnen Schichten unabhängig voneinander und gaußförmig verteilt sind. [Ama81] Der Rauschterm entsteht

durch elektrisches Rauschen in der Signalverarbeitung und der Auslese des Detektors. [Fab03] Der konstante Term kommt durch energieunabhängige Effekte, wie Inhomogenitäten im Aufbau des Detektors, Ungenauigkeiten in der Anfertigung, Temperaturgradienten oder Strahlungsschäden zustande. [Fab03]

3.3.2. Hadronische Kalorimeter

Da ein Teil der in einem hadronischen Schauer deponierten Energie nicht nachweisbar ist, liefert ein Kalorimeter im Allgemeinen ein kleineres Signal für Hadronen als für Elektronen. [Fab03] Eine quantitative Beschreibung erfolgt durch das Verhältnis e/h , das somit im Allgemeinen größer als eins ist. [Wig00]

Ein Kalorimeter, das die gleichen Signale für ein Hadron und ein Elektron liefert und damit ein Verhältnis von $e/h = 1$ aufweist, wird als kompensierend bezeichnet. [Wig00] Die Kompensation ist eine innere Eigenschaft eines Kalorimeters [Kol16] und kann nicht direkt gemessen werden [Wig00]. Die Bestimmung des e/h -Verhältnisses erfolgt durch die Messung des e/π -Verhältnisses. [Wig00] Das e/π -Verhältnis gibt das Verhältnis der Signale eines Elektrons und eines Pions an und ist definiert durch

$$\frac{e}{\pi} = \frac{e/h}{1 - f_{em}(1 - e/h)}. \quad [Wig00] \quad (3.13)$$

Daher hängt das e/π -Verhältnis über den elektromagnetische Schaueranteil f_{em} von der Energie des einfallenden Pions ab. Das e/π -Verhältnis wird für große Energien unabhängig von der Kompensation des Kalorimeters und strebt gegen eins.

Die Kompensation verbessert die Linearität und die Auflösung eines hadronischen Kalorimeters. [Wig00, Kol16, Fab03] Die Antwort nicht-kompensierender Kalorimeter ist nicht linear, da sich mit größer werdender Energie des einfallenden Teilchens der elektromagnetische Anteil des Schauers vergrößert. Da die elektromagnetische Komponente ein höheres Signal liefert, ist die Antwort eines nichtkompensierenden Kalorimeters auf Teilchen höherer Energie größer. Auch die Auflösung des Kalorimeters verbessert sich, falls Kompensation vorliegt. Der Anteil der elektromagnetischen Schauerkomponente fluktuiert stark. Ist das Kalorimeter nicht kompensierend, so entstehen dadurch von Ereignis zu Ereignis unterschiedlich große Signale und die Auflösung verschlechtert sich.

Daher ist die Kompensation ein Konstruktionskriterium hadronischer Kalorimeter. Kolonoski und Wermes [Kol16] geben eine Möglichkeit zur Einstellung des Verhältnisses $e/h = 1$ an. Im Allgemeinen gilt $e/h > 1$. Daher führt eine Verminderung des elektromagnetischen Signals unter gleichzeitiger Erhöhung des hadronischen Signals zur Kompensation. Die Verminderung des elektromagnetischen Signals kann durch die Verwendung von Absorbermaterialien mit hohen Kernladungszahlen erreicht werden. Ein Großteil der Energiedeposition elektromagnetischer Schauer findet durch Absorption niederenergetischer Photonen im Absorber statt. In diesen Prozessen werden Elektronen frei, die in Absorbern mit hohen Kernmassen das aktive Medium nicht mehr erreichen können und daher nicht mehr zum Signal beitragen können. Die Vergrößerung des hadronischen Anteils erfolgt durch den verbesserten Nachweis kalter Evaporationsneutronen. Der Energieübertrag von Neutronen ist inversproportional zur molaren Masse des Materials. Deshalb durchqueren Neutronen das passive Medium ohne Energie zu verlieren und übertragen ihre Energie im aktiven Medium über elastische Streuung an Protonen. Diese Protonen besitzen eine kurze Reichweite und erreichen daher das passive Medium nicht mehr. Die Erhöhung des Signals der nuklearen Komponente des Schauers kann damit durch die Variation der Schichtdicken von aktivem und passiven Medium gegeneinander oder durch das Anreichern des aktiven Mediums mit Wasserstoff erreicht werden.

4. Das Paket `geant4_10_3_package`

Der Zweck dieses Kapitels ist es, den Aufbau der Anwendung darzustellen und zu begründen. Zunächst erfolgt eine Analyse der Anforderungen, die an die Anwendung gestellt werden. Daran schließt sich die Beschreibung des Aufbaus des Pakets und der Möglichkeiten der Visualisierung an. Das folgende Unterkapitel diskutiert die Voraussetzungen an die Computerumgebung. Den Abschluss des Kapitels bildet die Beschreibung der Nutzung des Pakets.

4.1. Konzeption

Die zu erstellende Anwendung dient dazu, Studien zur Detektorsimulation und der Wechselwirkung von Teilchen mit einem Detektor durchzuführen. Die Durchführung dieser Studien erfolgt im Rahmen des Computerpraktikums zur Vorlesung “Teilchenphysik I“ und zur Vorlesung “Detektoren für Teilchen- und Astroteilchenphysik“. Denkbar ist auch eine Verwendung einfacher Analysen schon im Bachelorstudium.

Im Vordergrund der Übungen steht das Kennenlernen sowie das Erlernen des Umgangs mit dem Werkzeugkasten Geant4. Da Geant4 zur Verwendung in der Forschung entwickelt wurde, stellt es einen Umfang an Funktionalität zur Verfügung, der eine direkte Verwendung im Rahmen des Computerpraktikums zu aufwändig gestaltet. Daher ist es notwendig, für die Übungen eine Anwendung zur Verfügung zu stellen, die das Kennenlernen von Geant4 im Rahmen des Praktikums ermöglicht. Die Anforderung an die Anwendung ist es, einen Kompromiss zwischen der Nähe zu einer Anwendung in Geant4 und der Abstraktion und Vereinfachung der Verwendung zu finden. Die Nähe zu Geant4 ist notwendig, um den Ablauf einer Simulation und die verwendeten Klassen kennen und anwenden zu lernen. Um dies zu gewährleisten, ist ein gut strukturierter und einfach nachvollziehbarer sowie gut dokumentierter Programmcode notwendig. Zusätzlich bietet es sich an, die Visualisierung des Detektors und der Ereignisse automatisch zu gestalten, um die Anwendung einfach zu halten.

Die Anwendung, die bisher für Studien mit Geant4 genutzt wurde, stammt aus dem Jahr 2008. Die Steuerung dieser Anwendung erfolgt über eine von der Klasse `TVirtualMC[TVi]`, die im Framework *ROOT* [Bru97] definiert ist, vererbten Klasse. Die indirekte Nutzung von Geant4 über die *ROOT*-Klasse führt dazu, dass der Ablauf der Anwendung nicht mehr mit dem Ablauf einer Geant4-Anwendung übereinstimmt. Weiterhin lädt die Anwendung viele Bibliotheken und verwendet viele Dateien, in denen sich die Definitionen der verwendeten Klassen befinden. Dies und die Tatsache, dass keine Dokumentation der Anwendung existiert, führt dazu, dass die Anwendung und ihr Ablauf unübersichtlich werden und macht es schwer, die Grundstruktur nachzuvollziehen. Um die Anforderungen, die an die Anwendung gestellt werden, zu erfüllen, ist es daher sinnvoll, die Steuerung der Anwendung unabhängig von *ROOT* zu gestalten.

Da durch die Programmbibliothek `Geant4Py` bereits eine Schnittstelle von Geant4 zur Programmiersprache *Python*[Pyt] existiert, bietet es sich an die neue Anwendung in der

Programmiersprache *Python* zu schreiben. Eine Abkehr von der Programmiersprache C++ beim Schreiben der Anwendung ist sinnvoll, da die im Vergleich zu *Python* kompliziertere Syntax sowie das Verwenden vieler Dateien zur Klassendefinition in C++ dazu führt, dass der Programmcode unübersichtlich wird und von der Implementierung ablenken kann. Ein weiterer Vorteil, den die Verwendung von *Python* bietet, ist die Möglichkeit die Anwendung über den *Python*-Interpreter interaktiv zu steuern.

Die Simulationen, die mit Hilfe der Anwendung durchführbar sind, reichen von der einfachen Betrachtung der Wechselwirkung eines Teilchens mit Materie bis zur Simulation eines kompensierenden Kalorimeters. Um diese verschiedenen Aufgaben bewältigen zu können, ist die Anwendung flexibel zu konzipieren. Sie sollte weiterhin die Möglichkeit bieten, Anwendungsteile, die nicht von jeder Simulation benötigt werden, einfach auszutauschen. Daher ist ein modularer Aufbau der Anwendung sinnvoll. Ein weiterer Aspekt, der für den modularen Aufbau spricht, ist, dass sich der modulare Charakter[Ago03] von Geant4 sich in die Anwendung fortsetzt. Aus diesen Gründen bietet sich der Aufbau in einem *Python*-Paket an.

4.2. Aufbau

Das Paket gliedert sich in das Subpaket `geometry` und die Submodule `materials`, `actions`, `libphyslist` und `vis`.

Im Subpaket `geometry` befinden sich die Module, in denen die Geometrien definiert sind. In jedem Modul befindet sich die Definition einer von `G4VUserDetectorConstruction`-Klasse vererbten Klasse. Die Namen des Moduls und der Klasse sind identisch. Zur Verfügung stehen sieben Geometriedefinitionen. Weiterhin besteht die Möglichkeit der Definition neuer Geometrien. Die Verwendung dieser zusätzlichen Geometrien erfolgt analog zur Verwendung der bereits definierten Geometrien. Es ist allerdings zu beachten, dass die Geometrien nach dem gleichen Muster, wie die vordefinierten Geometrien geschrieben sind. Außerdem darf der Modul-, beziehungsweise der Klassenname nicht mit einem Unterstrich beginnen.

Die Definition aller Materialien, die im Detektor verwendet werden können, erfolgt im Submodul `materials`. Bereits definiert sind die Materialien Vakuum, Luft, Wasser, Eisen, Blei, flüssiges Argon, Szintillator, Polyethylen und Knochen. Die Eigenschaften der ersten drei Materialien und der Knochen stammen aus der NIST-Datenbank [Cou15] und die Eigenschaften des Szintillators stammen aus der *ROOT*-Anwendung. Die Daten der restlichen Materialien sind aus dem Review [Pat16] der Particle Data Group entnommen. In Geant4 enthält die Implementierung der `G4VUserDetectorConstruction`-Klasse die Definition der Materialien. Die Existenz des Untermoduls ist damit zu begründen, dass mehrere verschiedene Geometrien definiert sind und diese teilweise die gleichen Materialien verwenden. Außerdem vereinfachen die zur Verfügung gestellten Materialien die Definition neuer Geometrien, da keine neuen Materialien definiert werden müssen.

Die Definition dreier verwendeter zusätzlicher Klassen, die zur Analyse der Ereignisse dienen, befindet sich im Submodul `actions`. Zusätzlich ist die Klasse `PrimaryGenerator`, die von der `G4ParticleGun`-Klasse vererbt ist, definiert. Sie dient dazu, die Eigenschaften des primären Teilchens, wie Energie, Flugrichtung und Startpunkt, festzulegen und einen primären Vertex zu erzeugen. Die drei Klassen, die zur Analyse der erzeugten Ereignisse verwendet werden, sind von den Klassen `G4UserRunAction`, `G4UserTrackingAction` und `G4UserSteppingAction` vererbt. Die Klasse `TrackingAction` speichert am Anfang und am Ende jeder Spur die Informationen über die Spur ab, die von den Methoden `calo_readout()` und `get_x0OfFirstVertex()` zur Analyse des Ereignisses verwendet werden. Dieser Schritt ist notwendig, da eine Speicherung der Trajektorien der Teilchen über Geant4Py nicht

möglich ist. Die Klasse `SteppingAction` dient der Analyse der im Detektor deponierten Energie. Zu diesem Zweck speichert sie nach jedem Schritt den Mittelwert zwischen dem Anfangs- und dem Endpunkt des Schrittes sowie die in diesem Schritt deponierte Energie in drei Listen. Die Klasse `RunAction` findet zum Zurücksetzen der von den Klassen `TrackingAction` und `SteppingAction` erzeugten Listen zu Beginn jeden Laufes Verwendung.

Das Submodul `libphyslist` stellt zwei zusätzlich definierte Physiklisten zur Verfügung, die Listen `MyQGSP_BERT` und `TherapyPhysics`. Beide sind von Referenzphysiklisten vererbt und erweitern deren Funktionalität. Zum Einen setzen sie die Mindestenergie für Neutronen auf 0,5 MeV. Fällt die kinetische Energie der Neutronen unter diesen Wert, so werden sie nicht mehr weiter durch den Detektor verfolgt und deponieren ihre gesamte Energie. Zum Anderen ermöglichen sie die Verwendung der `G4UserLimits`-Klasse. `MyQGSP_BERT` basiert auf der Physikliste `QGSP_BERT` und `TherapyPhysics` auf der `QGSP_BIC`-Liste. Beide verwenden die Standardliste der elektromagnetischen Wechselwirkungen und das parametrisierte Quark-Gluon-String-Modell für hochenergetische hadronische Wechselwirkungen.[All16] Sie unterscheiden sich nur in den Modellen der intranuklearen Kaskade. Die Liste `QGSP_BERT` verwendet die Bertini-Kaskade für diese Wechselwirkungen. Die Bertini-Kaskade ist für Energien zwischen 0 GeV und 15 GeV gültig. Die binäre Kaskade, die in der Liste `QGSP_BIC` Anwendung findet, liefert für Pionen mit Energien bis zu 1,3 GeV genauere Ergebnisse. Für Protonen und Neutronen ist sie im Bereich bis 10 GeV gültig.[All16]

4.3. Visualisierung

Das Submodul `vis` dient der Visualisierung der Detektorgeometrie und der während eines Ereignisses produzierten Spuren. Die Visualisierung der Detektorgeometrie sorgt dafür, dass Fehler in der Definition schnell gefunden werden können. Üblicherweise erfolgt die Definition der Visualisierungskomponenten in einem Makro.[Gea16a] Von dieser Konvention wird abgewichen, da eine Methode zum Zeichnen der Detektorgeometrie definiert wird. Diese dient einer möglichen Kontrolle der verwendeten Geometrie vor der Simulation des ersten Ereignisses.

Geant4 unterstützt 14 verschiedene Visualisierungstreiber, die unterschiedliche Stärken und Schwächen besitzen.[All12] Acht dieser Treiber stehen auch über Geant4Py zur Verfügung. Zu diesen gehören zwei Treiber, die auf der Bibliothek OpenGL basieren, und jeweils ein Treiber basierend auf HepRepFile und DAWNFILE. Zusätzlich zu graphischen Ausgaben besteht die Möglichkeit, den hierarchischen Aufbau des Detektors in einer Textdatei zu beschreiben. Diese stellt der ASCII-Tree-Treiber bereit.[Gea16a]

Zur Visualisierung dient der Treiber `OpenGLImmediateX`. Er ist der einfachste der zur Verfügung gestellten Visualisierungstreiber und bietet im Gegensatz zum zweiten auf OpenGL basierenden Treiber, `OpenGLStoredX`, den Vorteil, dass einzelne Ereignisse gezeichnet werden können. `OpenGLStoredX` überlagert neu erzeugte Ereignisse mit bereits gezeichneten.

Das Einzeichnen der Geometrie des Detektors in das Fenster, das beim Import des Submoduls erzeugt wurde, findet während der Initialisierungsphase statt. Nach jedem simulierten Ereignis erfolgt die Aktualisierung des Detektors und das Einzeichnen der erzeugten Spuren. Um die Laufzeit aufwändiger Simulationen zu verkürzen, besteht die Möglichkeit die Aktualisierung der Visualisierung zu unterdrücken. Zusätzlich besteht die Möglichkeit, allgemeine Visualisierungsobjekte, wie Texte und Koordinatensysteme einzuzeichnen.

Die Farbe und Form der Darstellung der Spuren lässt sich durch verschiedene Modelle festlegen.[All16] Standardmäßig erfolgt die Farbgebung der Teilchenspuren auf Basis der Ladung des Teilchens. Verschiedene Eigenschaften der Teilchen können genutzt werden,

Tabelle 4.1.: Farbgebung der Trajektorien der verschiedenen Teilchen.

Teilchen	Farbe
Photon	grün
Lepton	blau
Antilepton	magenta
Neutrino/Antineutrino	rot
Proton, π^+ , K^+ , α -Teilchen	rot
Antiproton, π^- , K^-	cyan
Neutron	grau

um dieses Verhalten zu überschreiben. Dies sind der Typ oder das Ursprungsvolumen eines Teilchens.[All16] Das Paket `geant4_10_3_package` verwendet den Teilchentyp, um die Farbe der Spur festzulegen. Die Farben der einzelnen Teilchen sind in Tabelle 4.1 dargestellt. Die Farbe nicht aufgeführter Teilchen ist grau. Die Visualisierung eines Ereignisses ist in Abbildung 4.1 gezeigt.

4.4. Installation

Die Entwicklung des Pakets `geant4_10_3_package` zielt auf die Verwendung im Rahmen des Computerpraktikums ab. Das Ziel ist es daher eine möglichst einfache Verteilung des Pakets zu gewährleisten. Geant4 liegt in der Form eines Quellcodes vor und muss vor der Verwendung kompiliert werden. Da das Kompilieren des Quellcodes mit etwas Aufwand verbunden ist und je nach System relativ viel Zeit beansprucht, stehen zwei bereits kompilierte Versionen der Geant4-Bibliotheken zur Verfügung.

Die erste Version der Bibliotheken ist auf den Rechnern im Poolraum der Fakultät für Physik vorhanden. Diese Version dient der Durchführung der Übungen im Rahmen des Computerpraktikums. Zur Verwendung im Poolraum steht daher das Archiv `geant4_exercise_pool.tar.gz` zur Verfügung, in dem sich das Paket, sowie ein Skript zum Aufsetzen der Umgebung und zur Durchführung der einzelnen Übungen befinden.

Studierenden, die die Übungen zu Hause lösen möchten, stehen zwei weitere Möglichkeiten der Verwendung zur Verfügung. Die einfachere und schnellere Variante ist die Verwendung der virtuellen Maschine zur Datenanalyse.[Qua] Für sie existiert ebenfalls eine kompilierte Version der Geant4-Bibliotheken, die durch das Entpacken eines weiteren Archivs verwendet werden kann. Zusätzlich zu den kompilierten Bibliotheken von Geant4 sind vier weitere Bibliotheken erforderlich, die über ein zur Verfügung gestelltes Skript installiert werden können.

Zur Installation der Geant4-Bibliotheken unter Linux steht ein Installationsskript zur Verfügung. Um einen reibungslosen Ablauf zu gewährleisten, sollte das System zunächst um alle Programme, die in der virtuellen Maschine installiert sind, ergänzt werden. Im Anschluss müssen die vier zusätzlichen Bibliotheken sowie eine Version der Software CMake[cm] installiert werden. Diese dient zum Kompilieren der Geant4-Bibliotheken.

4.5. Verwendung

Mit dem Import des Pakets `geant4_10_3_package` werden das Subpaket `geometry`, alle Module, die sich in diesem Subpaket befinden, sowie alle Submodule importiert. Im Zuge des Imports erfolgt die Erstellung aller Materialien und die Definition aller Einstellungen für die Visualisierung und das Fenster, in dem die Visualisierung stattfindet, wird erzeugt.

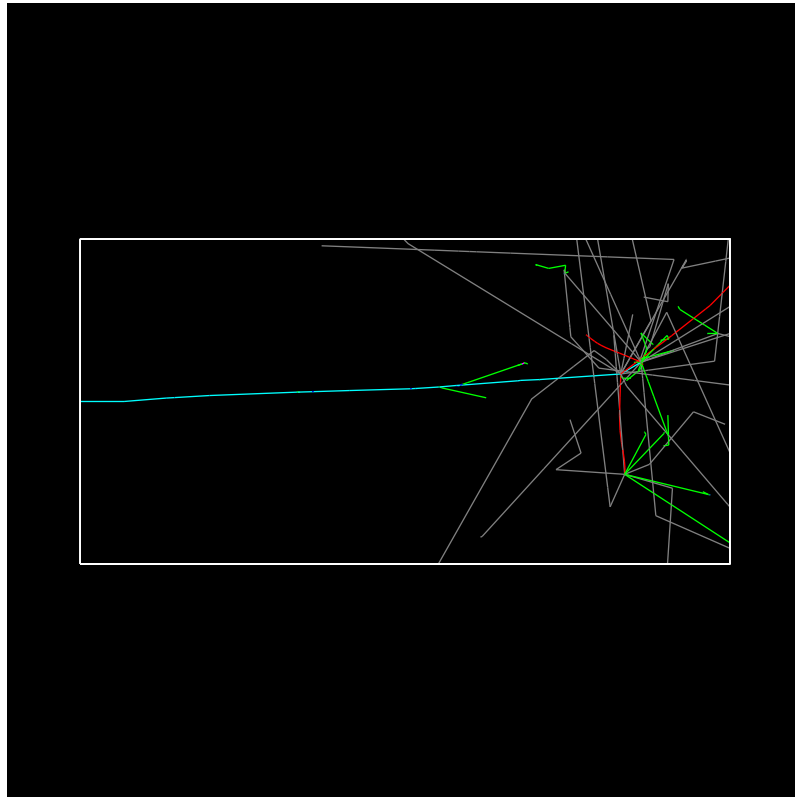


Abbildung 4.1.: Darstellung eines Ereignisses Zu sehen sind die Vielfachstreuung und die Absorption eines negativ geladenen Pions in einem Bleiquader. Die drei roten Spuren stammen von erzeugten Protonen.

Die Struktur des Ablaufs einer Anwendung entspricht der in Kapitel 2.1 diskutierten Struktur. Eine Instanz der `ApplicationManager`-Klasse kapselt die Aufrufe der `RunManager`-Methoden und führt so zu einer größeren Abstraktion und einfacheren Verwendung von Geant4. Dabei bleibt die Struktur der Anwendung erhalten. Nach dem Import des Pakets befindet sich Geant4 im `preInit`-Zustand. Zu Beginn einer Simulation muss eine Instanz der `ApplicationManager`-Klasse erzeugt werden. Die Übergabe der drei erforderlichen Klassen und der drei zusätzlich verwendeten Aktionsklassen an den `RunManager` erfolgt durch den Aufruf der `initialize()`-Methode der `ApplicationManager`-Klasse. Vor der Initialisierung besteht die Möglichkeit, die verwendete Detektorgeometrie und die Physikliste festzulegen. Zur Verfügung stehen die beiden im Submodul `libphyslist` definierten Physiklisten und die Referenzphysiklisten, die Geant4Py zur Verfügung stellt. Die drei zusätzlichen Aktionsklassen sowie eine Instanz der `G4ParticleGun`-Klasse, zur Erzeugung primärer Teilchen und Vertices, verwendet jede Anwendung. Nach erfolgreicher Initialisierung stehen verschiedene Methoden zur Einstellung der Eigenschaften des primären Teilchens zur Verfügung. Sind alle gewünschten Einstellungen vorgenommen, erfolgt der Start eines Laufes über die `start_run()`-Methode. Zusätzlich ermöglicht sie die Unterdrückung der Visualisierung des simulierten Laufs. Zwischen zwei Läufen besteht die Möglichkeit, den Aufbau des Detektors zu verändern. Dies erfolgt durch das Festlegen des neuen Aufbaus und einen anschließenden erneuten Aufruf der `initialize()`-Methode. Die Änderung der Physikliste nach der Initialisierung ist nur durch einen Neustart der Anwendung möglich.

Zwischen zwei Läufen besteht die Möglichkeit der Analyse der simulierten Ereignisse. Zu diesem Zweck stehen die Methoden `calo_readout()`, `get_x0fFirstVertex()` und `get_edep_in_volume()` zur Verfügung. Die `calo_readout()`- und `get_x0fFirstVertex()`-Methode verwenden die Informationen, die durch die `TrackingAction`-Klasse gespeichert

wurden. Aus den durch die Klasse `SteppingAction` gespeicherten Informationen lassen sich Histogramme der Verteilung der Energiedeposition erzeugen. Die Methode `calo_readout()` bestimmt die Anzahl an geladenen Spuren im aktiven Medium eines Kalorimeters. Sie verwendet die Volumina, in denen der Start- und der Endpunkt einer Spur liegen, und berechnet daraus die Anzahl der durchquerten aktiven Schichten. Die Energie, die in einem bestimmten Bereich des Detektors deponiert wurde, berechnet die `get_edep_in_volume()`-Methode. Die Methode `getX0fFirstVertex()` liefert die x-Koordinate des Ortes, an dem die erste Wechselwirkung des primären Teilchens stattfindet. Sie überprüft zunächst, welche Spuren durch das primäre Teilchen erzeugt wurden, und sucht anschließend die minimale x-Koordinate der Entstehungsorte der Teilchen.

5. Beispielsimulationen

Dieses Kapitel dient der Zusammenstellung möglicher Simulationen und deren Ergebnisse. Im Verlauf des Kapitels werden Simulationen besprochen, mit denen die in Kapitel 3.3.2 vorgestellten Grundlagen simuliert werden. Diese Simulationen dienen sowohl der Verifikation der Auslesefunktionen als auch der Darstellung möglicher Aufgaben im Praktikum zu den Teilchenphysikübungen. Zunächst erfolgt für jede Simulation eine Beschreibung der verwendeten Geometrie, Physikliste sowie sonstiger Besonderheiten der Simulation. Der erste Teil des Kapitels beschäftigt sich mit elektromagnetischen Wechselwirkungen. Hierbei erfolgt die Simulation der Absorption von Photonen und der Entwicklung elektromagnetischer Schauer. Im nächsten Abschnitt erfolgt die Diskussion der simulierten Eigenschaften eines elektromagnetischen Kalorimeters. Der dritte Abschnitt beschäftigt sich mit einem kompensierenden Kalorimeter und der letzte Abschnitt diskutiert die Anwendung des Bragg-Peaks in der Strahlungstherapie. Bei den meisten der angesprochenen Aufgaben und Analysen handelt es sich um eine Neuimplementierung der alten Übungsaufgaben über das entwickelte Paket.

5.1. Elektromagnetische Wechselwirkungen

Bestimmung der Strahlungslänge aus der Absorption von Photonen

Die Bestimmung der Strahlungslänge aus dem Absorptionsverhalten von Photonen, die Paarbildungen induzieren, dient der Einführung in die Verwendung des Pakets sowie einem Test der Methode `get_x0ffirstVertex()`. Ziel der Analyse ist es, die Absorption von Photonen zu untersuchen und aus dem Absorptionsverhalten die Strahlungslänge X_0 zu bestimmen.

Die Simulation verwendet den Detektor `pbbox`. Er besteht aus einem Bleiquader mit Abmessungen, die der Nutzer bei der Übergabe der Detektorgeometrie an den `ApplicationManager` festlegt. Da die erste Wechselwirkung in der Regel schon nach wenigen Zentimetern stattfindet, genügt es, einen Würfel mit einer Seitenlänge von 4 cm zu betrachten. Die Verwendung eines kleinen Detektorvolumens bietet den Vorteil, dass sich nur ein kleiner Teil des entstehenden Schauers im Detektorvolumen befindet. Dies führt zu einer Verringerung des Simulationsaufwands und damit zu einer kürzeren Laufzeit des Programms. Da in diesem Fall keine Begrenzung der Schrittlänge notwendig ist, findet die Referenzphysikliste `QGSP_BERT` Verwendung.

Während der Simulation werden Photonen mit einer kinetischen Energie von 1 GeV einzeln erzeugt und in den Quader geschossen. Die Energie der Photonen ist damit so hoch, dass nur Paarbildungsprozesse stattfinden. Abbildung 5.1 zeigt die Anzahl der aufgezeichneten Ereignisse über der Eindringtiefe des Photons. Die erhaltene Verteilung deckt sich mit dem aufgrund des Absorptionsprozesses erwarteten exponentiellen Abfall. Die große Anzahl an Ereignissen im letzten Bin ist keine statistische Fluktuation, sondern entsteht durch die Auslesefunktion, die die kleinste x-Koordinate eines Vertices sucht. Durchqueren Photonen

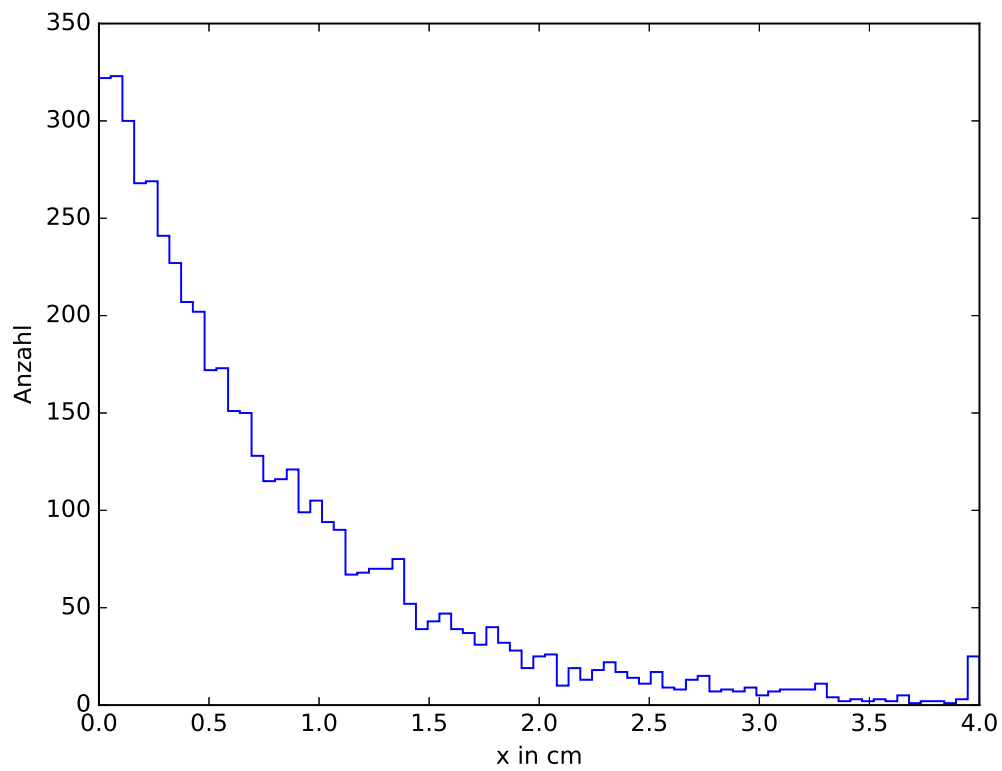


Abbildung 5.1.: Verteilung des Ortes der ersten Paarbildung Dargestellt ist die Anzahl der Vertices über der Strecke, die die Teilchen im Detektor zurückgelegt haben. Die Gesamtzahl der simulierten Ereignisse beträgt 5000. Erzeugt mit *matplotlib*[Hun07].

Tabelle 5.1.: Ergebnisse der Messung der Strahlungslänge. Der Mittelwert und die Standardabweichung von 10 Messungen sind den Literaturwerten aus [Pat16] für die entsprechenden Materialien gegenübergestellt.

Material	X_0 in g cm^{-2}	$X_{0,\text{Lit}}$ in g cm^{-2}
Pb	$6,47 \pm 0,19$	6,37
Fe	$14,16 \pm 0,18$	13,84

den Bleiquader ohne zu wechselwirken, so ist das Minimum aller Vertices die Länge des Detektors.

Aus der erhaltenen Verteilung der primären Vertices lässt sich die Strahlungslänge bestimmen. Nach dem Absorptionsgesetz, welches in Gleichung 3.6 angegeben ist, beträgt die Intensität eines Photonenstrahles nach Durchqueren einer Strahlungslänge noch 46 %. Damit sind 54 % der Photonen im Strahl schon absorbiert worden. Durch Bestimmen des Ortes, bis zu welchem 54 % der Wechselwirkungen stattgefunden haben, lässt sich die Strahlungslänge bestimmen. Die für die Materialien Blei und Eisen bestimmten Strahlungslängen sind in Tabelle 5.1 angegeben. Beim Vergleich der gemessenen Werte mit den Literaturwerten stellt sich heraus, dass die gemessenen Werte größer sind, als die Literaturwerte. Die gemessene Strahlungslänge von Blei stimmt im Rahmen der Standardabweichung mit dem Literaturwert überein. Für Eisen liegt der gemessene Wert etwas zu hoch.

Elektromagnetische Schauer

Die Bestimmung der Länge und Breite eines elektromagnetischen Schauers sowie deren Energieabhängigkeit dient der Einführung in die Verwendung der Listen, die durch die `SteppingAction`-Klasse erzeugt wurden.

Als Detektor für die Analysen der Dimensionen eines elektromagnetischen Schauers fungiert wieder der Blei-Quader. Allerdings erfolgt die Wahl der Größe so, dass der Bleiquader den ganzen Schauer enthält. Aus diesem Grund beträgt die Länge des Quaders in diesem Fall 40 cm. Die Höhe und Breite des Quaders werden zu 30 cm gewählt. Die Analyse der Länge und der Breite des Schauers erfolgt durch die Verteilung der deponierten Energie im Detektor. Dabei wird der gesamte Energieverlust pro Simulationsschritt über dem Mittelwert des Anfangs- und des Endpunktes des Schrittes aufgetragen. Daher hängt die Genauigkeit der Verteilung der Energie von der Länge eines Simulationsschrittes ab. Je länger der Schritt ist, desto größer wird die Differenz zwischen dem Ort des Energieverlustes und dem Ort, der in der Liste gespeichert wird. Deshalb kommt in der Simulation eine zusätzliche Begrenzung der maximal erlaubten Schrittlänge auf 4 mm zur Anwendung. Die Integration der maximal erlaubten Schrittlänge in die Anwendung erfolgt durch die Physikliste `MyQGSP_BERT`.

Die Abbildung 5.2 zeigt die Verteilung der gemessenen Längen und Breiten von hundert Schauern. Bei den vermessenen Schauern handelt es sich um Schauer, die von einem Elektron der Energie 1 GeV induziert wurden. Die aufgetragene Länge ist die Länge, innerhalb der 95 % der Energiedeposition liegen. Die aufgetragene Breite entspricht dem Molière-Radius.

Um die Energieabhängigkeit der Länge und Breite elektromagnetischer Schauer zu bestimmen, werden Elektronen mit Energien von 1 GeV bis 15 GeV verwendet. Die Mittelwerte der Länge und Breite zehn simulierter Schauer je Energie sind in Abbildung 5.3 dargestellt. Die obere Abbildung zeigt die Energieabhängigkeit der Länge elektromagnetischer Schauer. Diese stimmt mit der erwarteten logarithmischen Zunahme der Länge eines Schauers mit

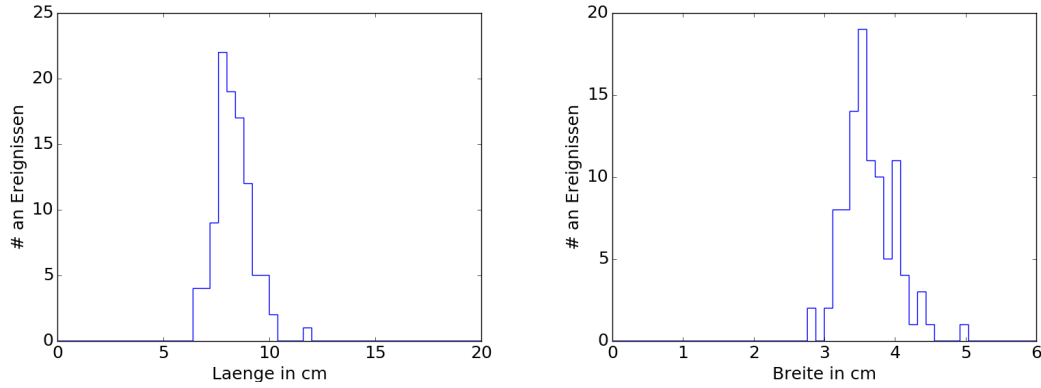


Abbildung 5.2.: Länge und Breite eines elektromagnetischen Schauers. Gezeigt ist die Verteilung der Länge und Breite von 100 simulierten Schauern. Plot erzeugt mit *matplotlib*[Hun07].

der Energie überein. Die Formel

$$t_{95\%} = \ln\left(\frac{E}{\epsilon}\right) + 0.008Z + 9,1 \quad (5.1)$$

gibt die Länge an, innerhalb derer 95 % der Energie eines Schauers deponiert werden.[Fab03] Die Angabe der Länge erfolgt in Einheiten der Strahlungslänge X_0 . ϵ ist die kritische Energie und Z die Ladungszahl des Materials. Die untere Abbildung zeigt den Verlauf des Molière-Radius mit der Energie. Auch im Fall des Molière-Radius stimmen die simulierten Werte mit der Erwartung überein. Ein konstanter Fit an die Daten ergibt einen Wert von

$$R_M = (1,62 \pm 0,03) \text{ cm} \quad (5.2)$$

für den Molière-Radius in Blei. Dieser Wert stimmt mit dem Literaturwert von 1,6 cm, der aus [Wig00] entnommen ist, überein.

5.2. Studien an einem elektromagnetischen Kalorimeter

Die Analyse der Antwort eines elektromagnetischen Sampling-Kalorimeters dient dem Kennenlernen der Funktionsweise eines Kalorimeters und der Grundlagen, die bei der Planung des Aufbaus eine Rolle spielen. Zu diesem Zweck bietet sich die Bestimmung der Kalibrationskonstanten sowie die Bestimmung der Auflösung in Abhängigkeit der Energie an. Der Vergleich der erhaltenen Ergebnisse mit der Erwartung dient als Test der Methode `calo_readout()`, die die Anzahl an Spuren geladener Teilchen im aktiven Medium zählt.

Die Untersuchungen werden an einem Sampling-Kalorimeter durchgeführt, dessen passive und aktive Schichten aus Blei und flüssigem Argon bestehen. Die Schichtdicke der passiven Lagen beträgt 2 cm, die der aktiven 1 cm. Das Kalorimeter enthält 16 passive und 15 aktive Lagen. Eine Begrenzung der Schrittweite der Simulation ist für die Untersuchungen nicht notwendig, daher findet die Physikliste `QGSP_BERT` Verwendung.

Das Signal, das zur Bestimmung der Kalibrationskonstanten dient, stammt von Elektronen kinetischer Energien von 1 GeV bis 15 GeV. Der Mittelwert und die Standardabweichung des Signals von 50 Ereignissen für jeden betrachteten Wert der Energie sind in Abbildung 5.4 über der Energie des Elektrons aufgetragen. Gezeigt ist zusätzlich der Fit einer Geraden an die Messwerte. Aus der Anpassung ergibt sich ein Wert von

$$m = (48,8 \pm 1,3) \text{ GeV}^{-1} \quad (5.3)$$

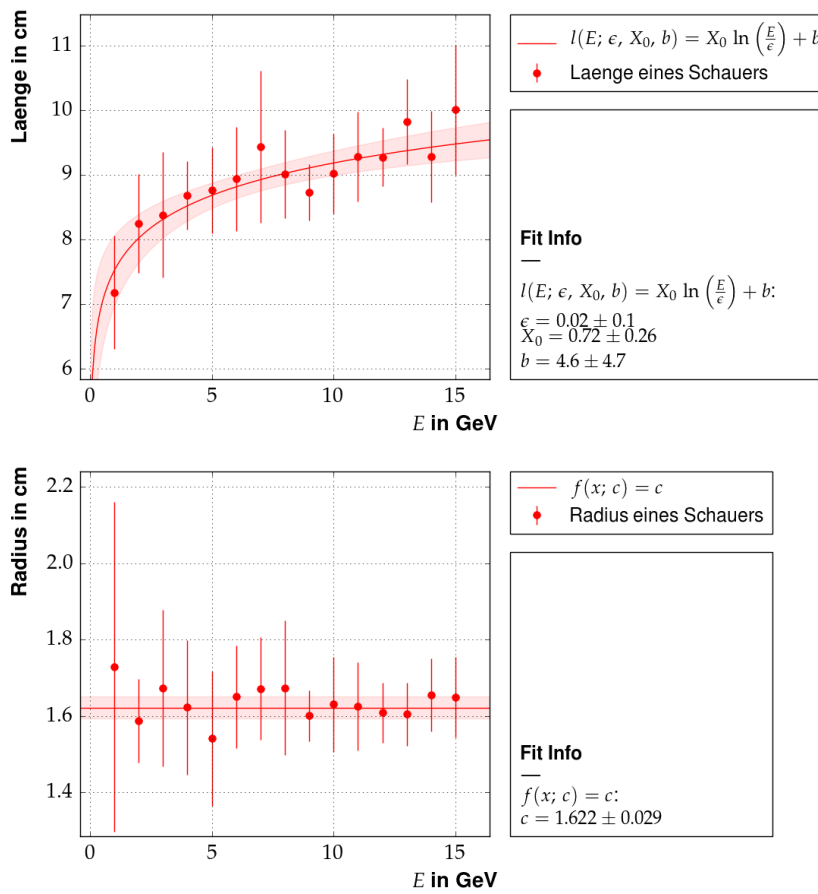


Abbildung 5.3.: Energieabhängigkeit der Länge und Breite elektromagnetischer Schauers. Die obere Abbildung zeigt die Energieabhängigkeit der Länge des Schauers sowie eine Anpassung der Daten an Gleichung 5.1. Die untere Abbildung stellt den konstanten Radius der Schauers dar. Plots erzeugt mit *kafe*[Säv13].

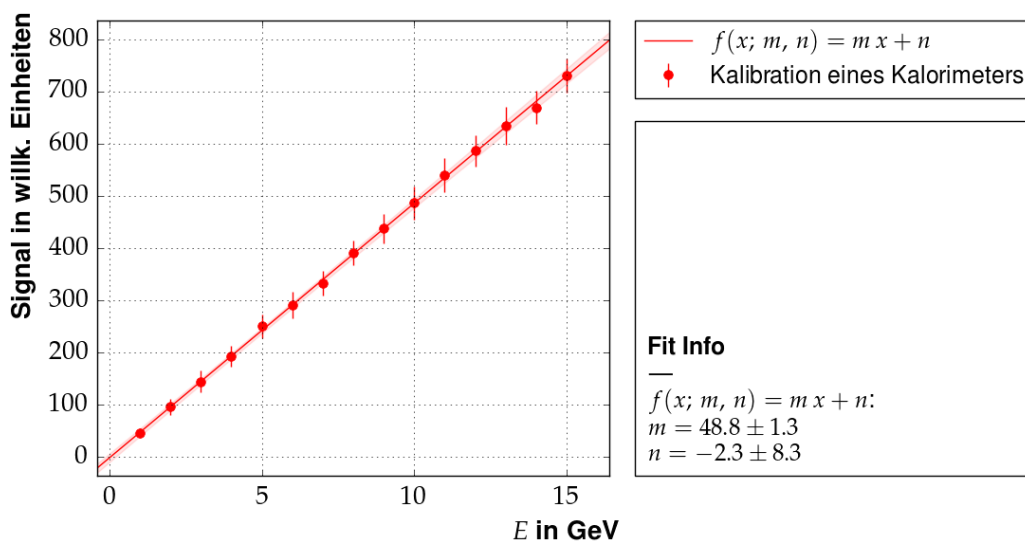


Abbildung 5.4.: Bestimmung der Kalibrationskonstante. Der Plot wurde mit *kafe*[Säv13] erzeugt.

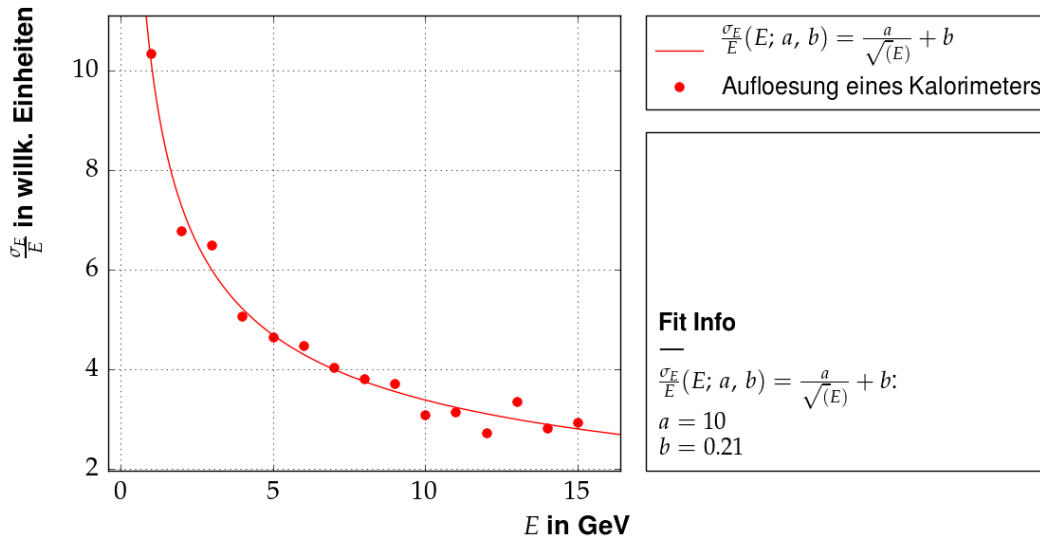


Abbildung 5.5.: Energieabhängigkeit der Auflösung eines elektromagnetischen Kalorimeters. Erzeugt mit *kafe*[Säv13].

für die Kalibrationskonstante des Kalorimeters. Die Linearität der erhaltenen Signale legt nahe, dass die Funktion zur Auslese des Kalorimeters korrekt funktioniert. Die erwartete Linearität der Signale lässt sich durch ein einfaches Schauermodell nach Heitler[Hei54] erklären. Dieses Modell geht davon aus, dass sie die Energie des einfallenden Elektrons bei der Wechselwirkung gleichmäßig auf ein erzeugtes Photon und das Elektron verteilt. Aus dem einfallenden Teilchen entstehen somit zwei Teilchen mit der Hälfte der Energie. Das Signal des einfallenden Teilchens ist somit doppelt so groß wie das Signal eines Teilchens der halben Energie.

Die Analyse der Auflösung des Kalorimeters betrachtet den statistischen Term der Auflösung, der in Kapitel 3.3.1 diskutiert wurde. Die Proportionalität dieses Terms ist durch

$$\frac{\sigma_E}{E} \propto \sqrt{\frac{t}{E}} \quad (5.4)$$

gegeben. Das Ziel der Analyse der Auflösung ist die Überprüfung dieser Relation. Hierbei liegt der Fokus auf der Energieabhängigkeit. Verwendet werden wieder Elektronen der Energien von 1 GeV bis 15 GeV.

Abbildung 5.5 stellt die Abhängigkeit der Auflösung zur Energie des einfallenden Elektrons dar. Als Auflösung ist die Standardabweichung der Anzahl der Spuren geladener Teilchen aus hundert Ereignissen dividiert durch die Energie der einfallenden Elektronen dargestellt. Zusätzlich zu den Messwerten ist die Anpassung der Messwerte an die Funktion

$$\frac{\sigma_N}{E} = \frac{a}{\sqrt{E}} + b \quad (5.5)$$

dargestellt und die Werte der Konstanten a und b sind angegeben.

5.3. Hadronisches Kalorimeter

Die Leistung eines hadronischen Kalorimeters hinsichtlich Auflösung und Linearität der Signale hängt vom Verhältnis der Signale eines Elektrons und eines Hadrons ab. Die beste Performance zeigen kompensierende Kalorimeter, die für Elektronen und Hadronen das gleiche Signal liefern.[Fab03] Diese Simulation dient der Optimierung des Signalverhältnisses von Elektronen und Pionen, dem e/π -Verhältnis. In Kapitel 3.3.2 wurde eine

Tabelle 5.2.: Studie zur Länge des Kalorimeters Die Länge l_{Fe} des Eisenquaders ist der Anzahl der Ereignisse mit einer Energiedeposition im Bleiquader gegenübergestellt. Die Werte für die Längen von 190 cm bis 250 cm sind in der Tabelle nicht aufgeführt, um die Tabelle kurz zu halten. Die Anzahl der Ereignisse mit Energieeintrag im Bleiquader für diese Längen beträgt null.

Länge in cm	Ereignisse
80	52
90	31
100	14
110	15
120	4
130	5
140	2
150	1
160	1
170	0
180	0

mögliches Verfahren zur Einstellung dieses Verhältnisses besprochen. Dieses beruht darauf, die Schichtdicken eines Kalorimeters gegeneinander zu variieren, bis das Kalorimeter kompensierende Eigenschaften zeigt.

Die Analyse des kompensierenden Kalorimeters gliedert sich in drei Simulationen. In der ersten Simulation erfolgt die Bestimmung der Länge, die das Kalorimeter besitzen sollte, um die Leckverluste möglichst gering zu halten. Das Ziel der zweiten Simulation ist es, das Verhältnis der Dicken von aktivem und passivem Medium zu finden, für das das e/π -Verhältnis möglichst nahe an eins ist. In der dritten und letzten Simulation wird das Verhalten des e/π -Verhältnisses für verschiedene Energien untersucht.

Die Untersuchungen werden an einem Sampling-Kalorimeter durchgeführt, das aus passiven Lagen aus Eisen und aktiven Lagen aus Szintillatoren besteht. Die Beschreibung der Eigenschaften des Kalorimeters erfolgt durch die `comp calor`-Klasse. Die Länge des Kalorimeters setzt sich aus der Gesamtlänge aller Schichten aus Eisen sowie der Gesamtlänge der aktiven Schichten zusammen. Die Gesamtlänge des Kalorimeters wird damit durch die Formel

$$L = l_{\text{Fe}} + l_{\text{Scint}} = l_{\text{Fe}} + \frac{d_{\text{Scint}}}{d_{\text{Fe}}} l_{\text{Fe}} \quad (5.6)$$

beschrieben. d_{Scint} gibt die Dicke einer aus Szintillatormaterial bestehenden Schicht und d_{Fe} die Dicke einer Eisenschicht an.

Zur Analyse der benötigten Gesamtlänge aller Schichten aus Eisen l_{Fe} steht die Geometriedefinition `lengthtest` zur Verfügung. Mit ihrer Hilfe wird die Länge eines Eisenquaders bestimmt, die ausreicht, um Pionen der Energie 3 GeV zu stoppen und den entstehenden Schauer möglichst ganz zu enthalten. Der Detektor besteht aus einem Eisenquader mit variabler Länge. Hinter dem Eisen befindet sich ein weiterer Quader, der aus Blei besteht. Der zweite Quader dient der Bestimmung der Anzahl der Ereignisse, deren Schauer nicht komplett im ersten Quader enthalten ist.

Die Ergebnisse der Simulation für 200 Ereignisse pro gewählter Länge sind in Tabelle 5.2 angegeben. Die Werte in der Tabelle zeigen einen starken Abfall mit zunehmender Länge des Eisenquaders. Da zur Gesamtlänge an Eisen im Kalorimeter noch die Länge aller aktiven Schichten hinzukommt, fällt die Wahl der Länge des Kalorimeters auf eine Länge

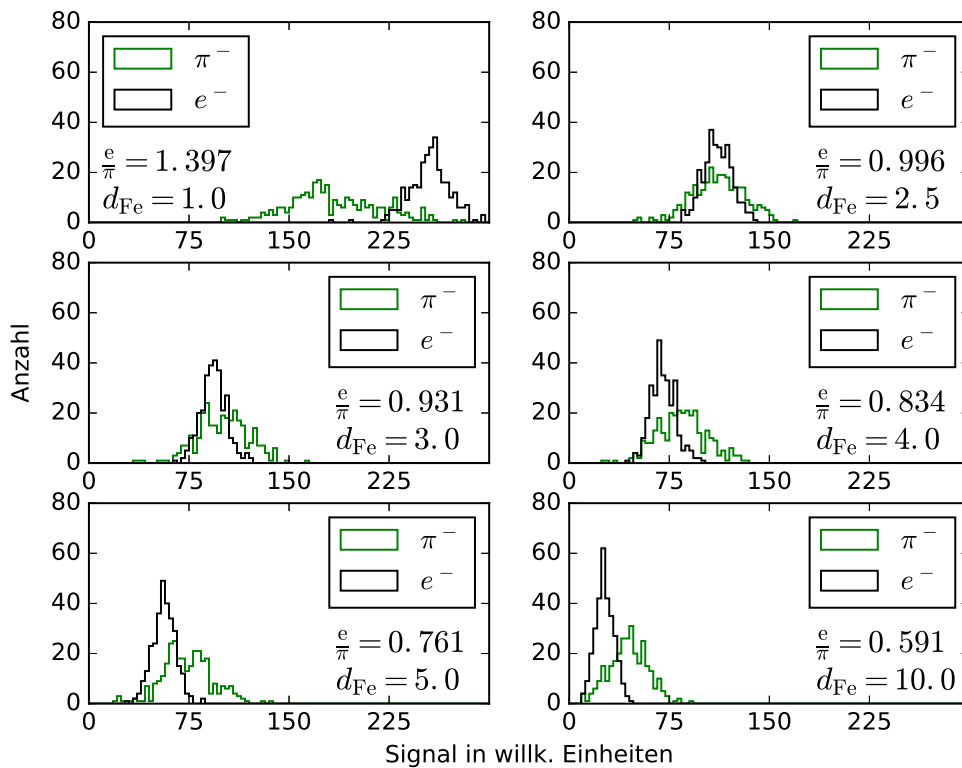


Abbildung 5.6.: Verteilung der Signale von Elektronen und Pionen. Gezeigt ist die Anzahl der registrierten Signale über der Höhe des Signals für die sechs untersuchten Schichtdicken. Die Schichtdicke des passiven Mediums in cm und das e/π -Verhältnis sind angegeben. Plots erzeugt mit *matplotlib*[Hun07].

von 150 cm. Dies ist die erste Länge, bei der der Anteil der Ereignisse mit Energiedeposition im Bleiquader weniger als 1% beträgt.

Zur Bestimmung des Verhältnisses der Schichtdicken, für das $e/\pi = 1$ gilt, werden die Signale von Elektronen und Pionen der Energie 3 GeV verwendet. Die Änderung des Verhältnisses der Schichtdicken erfolgt durch die Variation der Dicke der passiven Schichten zwischen 1 cm und 10 cm. Die Dicke der aktiven Schichten bleibt konstant bei 1 cm. Das e/π -Verhältnis berechnet sich aus dem Quotienten der Mittelwerte von jeweils 300 Elektronen- und Pionenereignissen.

Während der Simulation werden Teilchen, deren kinetische Energie unter 500 keV fällt, nicht mehr weiter verfolgt. Sie deponieren ihre verbliebene Energie an dem Ort, an dem diese Grenze erreicht wird. Der Grund, eine solche Beschränkung einzuführen, liegt an der Laufzeit des Programms. Durch die Beschränkung erzeugen Teilchen, die unter diese Energie fallen, keine weiteren sekundären Teilchen, die durch den Detektor verfolgt werden müssen. Deshalb verringert sich die Teilchenmultiplizität und damit die Laufzeit der Simulation. Allerdings kommt es durch die Beschränkung auch zu einer Änderung des Schichtdickenverhältnisses, für das Kompensation eintritt. Die elektromagnetische Komponente eines Schauers wird unterdrückt, da niederenergetische Teilchen, deren Erzeugung am Ende einer passiven Schicht stattfindet, nicht mehr in das aktive Medium gelangen, wenn ihre Energie unter 500 keV sinkt. Sie tragen deshalb nicht mehr zum Signal bei.

Abbildung 5.6 zeigt die Signale der Elektronen und Pionen für die verschiedenen Absorberdicken. Zu beobachten ist eine Verschiebung des Schwerpunktes der einzelnen Verteilungen

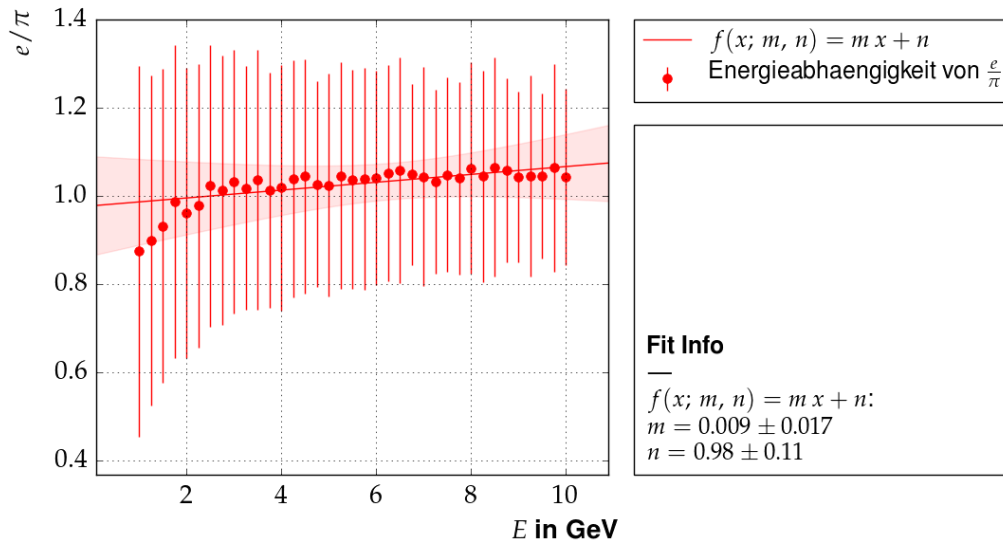


Abbildung 5.7.: Energieabhängigkeit des e/π -Verhältnisses. Für kleine Energien liegen die Ergebnisse unter einem Wert von 1. Im Energiebereich zwischen 2 GeV und 3 GeV beträgt das Verhältnis etwa 1 und für größere Energien steigt es weiter leicht an. Erzeugt mit *kafe*[Säv13].

gegeneinander. Eine passive Schichtdicke von $d_{\text{Fe}} = 2,5$ cm führt zum besten Ergebnis des e/π -Verhältnisses. Wird die Schichtdicke des passiven Mediums weiter vergrößert, so stellt sich eine Überkompensation der Signale von Pionen und Elektronen ein. Ein kleineres Verhältnis führt zu einer Unterkompensation. Weiterhin ist zu beobachten, dass die Höhe der Signale von Elektronen und Pionen mit zunehmender passiver Schichtdicke steigt. Dieser Effekt kommt dadurch zustande, dass durch die dickere passive Schichtdicke mehr Teilchen, deren Erzeugung im passiven Medium stattfindet, dieses nicht mehr verlassen können.

Den Abschluss der Untersuchungen an einem hadronischen Kalorimeter bildet die Analyse der Energieabhängigkeit des e/π -Verhältnisses. Die Untersuchung verwendet die passive Schichtdicke von 2,5 cm, für die das e/π -Verhältnis am nächsten an 1 lag. Die betrachteten Energien reichen von 1 GeV bis 10 GeV. Die Bestimmung des e/π -Verhältnisses erfolgt wieder durch die Simulation von 300 Ereignissen und Bilden des Mittelwertes der Signalverteilungen. Das erhaltene Ergebnis ist in Abbildung 5.7 gezeigt. Die Unsicherheiten der einzelnen Messwerte stammen aus der Fehlerfortpflanzung der Standardabweichung vom Mittelwert der Signale. Die Berechnung der Unsicherheiten erfolgt über die Formel

$$\Delta e/\pi = \frac{e}{\pi} \left(\frac{\Delta e}{e} + \frac{\Delta \pi}{\pi} \right) \quad (5.7)$$

Hierbei stehen e und π für die Mittelwerte der Signalverteilung und Δe und $\Delta \pi$ für die jeweilige Standardabweichung vom Mittelwert des Signals. Der Verlauf des e/π -Verhältnisses mit der Energie entspricht nicht ganz dem zu erwartendem Ergebnis von einer asymptotischen Annäherung an den Wert 1.

5.4. Bestrahlung eines Tumors

Die letzte der vorgestellten Beispielsimulationen beschäftigt sich mit der Simulation der Bestrahlung eines Tumors mit α -Teilchen. Diese ist die einzige der vorgestellten Simulationen, bei der es sich nicht um die Präsentation der Ergebnisse einer alten Aufgabe handelt. Zu Beginn der Simulation liegt der Fokus darauf, zu begründen, warum sich die

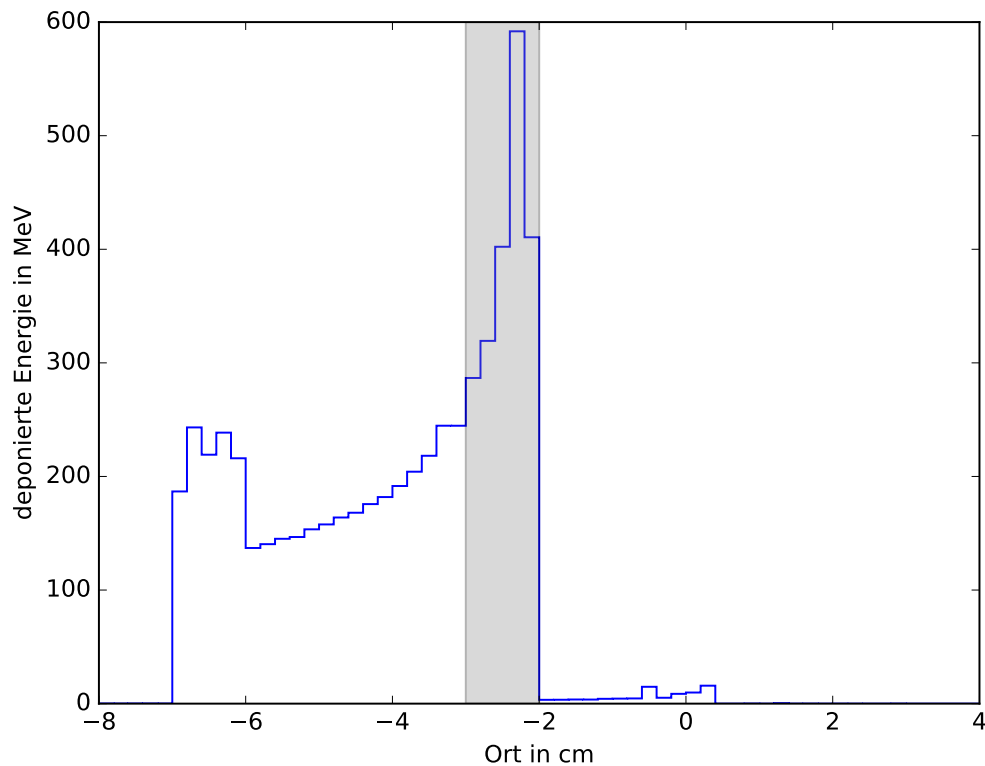


Abbildung 5.8.: Profil der Energiedeposition Dargestellt ist die gesamte Energiedeposition von 20 α -Teilchen der Energie 320 MeV. Der Bereich, in dem sich der Tumor befindet ist grau hinterlegt. Erzeugt mit *matplotlib*[Hum07]

Bestrahlung eines Tumors mit schweren geladenen Teilchen anbietet. Daran schließt sich eine Optimierung der im Tumor deponierten Energie an.

Die Beschreibung der Geometrie, die für diese Simulation verwendet wird, erfolgt über die Klasse `tumorthrapy`. Die Geometrie ist eine einfache Repräsentation eines Kopfes, der einen Tumor enthält. Der Schädel wird durch eine Kugelschale mit einer Dicke von 1 cm repräsentiert, die aus Knochen besteht. Das Gehirn sowie der Tumor werden durch Wasser enthaltende Vollkugeln dargestellt. Der gesamte Kopf besitzt einen Radius von 7 cm, während der Radius des Tumors 0,5 cm beträgt. Um eine möglichst genaue Auflösung des Ortes der Energiedeposition zu erhalten, unterliegt die maximale Schrittweite eines Simulationsschrittes einer Beschränkung. Sobald die α -Teilchen den Kopf erreichen, beträgt die maximale Schrittweite 2 mm. Erreichen die Teilchen den Tumor, wird die maximale Schrittweite auf 0,5 mm herabgesetzt. Die Simulation verwendet die Physikliste `TherapyPhysics`, da diese auf der Referenzphysikliste `QGSP_BIC` basiert, die häufig in medizinischen Anwendungen verwendet wird [All16], und die Integration der Schrittweitenbegrenzung ermöglicht.

Schwere Ionen verlieren ihre Energie durch die Ionisation des Mediums, das sie durchqueren. Ihre Energie, die sie auf ihrem Weg durch die Materie verlieren, ist damit durch die Bethe-Gleichung gegeben, die in 3.1.1 angegeben ist. Aus der Bethe-Gleichung lässt sich schließen, dass die Energieverluste der Teilchen größer sind, je kleiner ihre Energie wird. Dies führt dazu, dass die Teilchen kurz bevor sie zur Ruhe kommen die meiste Energie deponieren. Das Profil der Energiedeposition, das dadurch zustande kommt, ist in Abbildung 5.8 dargestellt. Zum Erzeugen der Abbildung 5.8 wurde der Energieverlust von 20 α -Teilchen der Energie 320 MeV beim Eindringen in die Geometrie des Kopfes simuliert. Der charakteristische

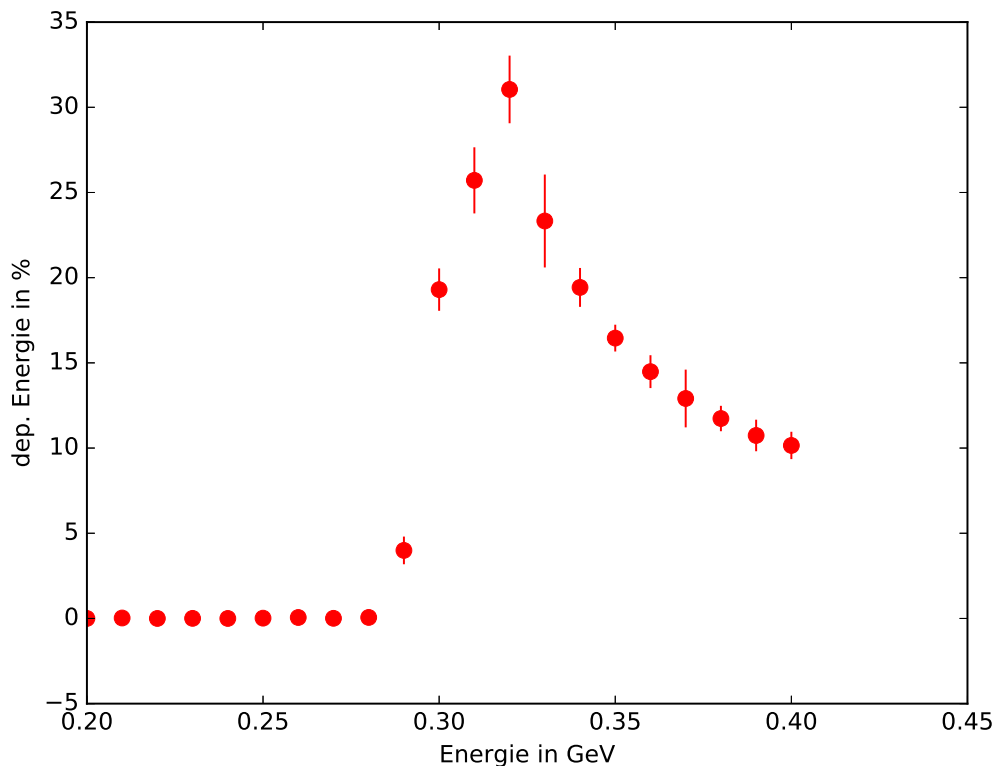


Abbildung 5.9.: Relative Energiedeposition im Tumor Die Abbildung stellt den Anteil der Energie dar, die im Tumor deponiert wurde. Aufgetragen sind die Mittelwerte sowie die Standardabweichung von zehn Simulationen für jeden Wert der Energie. Erzeugt mit *matplotlib*[Hun07].

Bragg-Peak am Ende der Reichweite der α -Teilchen ist deutlich zu erkennen. Auf den ersten Zentimetern, die die α -Teilchen in das Material eindringen, verlieren sie mehr Energie als im weiteren Verlauf. Die höhere Dichte des Knochens führt zu diesem Effekt. Im Wasser nimmt die Energiedeposition anschließend kontinuierlich zu, bis sie ihr Maximum am Bragg-Peak erreicht hat. Nach Erreichen des Bragg-Peaks fällt die Energiedeposition rasch ab. Die Energiedeposition, die an der Stelle $x = 0$ cm zu sehen ist, entsteht durch die Erzeugung sekundärer Teilchen, bevor der Bragg-Peak erreicht wird.[Kol16] Die erzeugten Teilchen sind leichter als das α -Teilchen und legen deshalb eine größere Strecke im Material zurück, bevor sie zur Ruhe kommen.

Ein Vorteil, den die Bestrahlung eines Tumors durch schwere geladene Teilchen mit sich bringt, ist die geringe Energiedeposition vor Erreichen des Bragg-Peaks. Dadurch wird das umliegende Gewebe im Vergleich zur Bestrahlung mit Elektronen oder Photonen geschont, die zu Beginn des Eindringens die meiste Energie deponieren. Ein weiterer Vorteil, den die Therapie mit schweren geladenen Teilchen mit sich bringt, ist die gute Einstellbarkeit der Reichweite der Teilchen.[Kol16]

Die zweite Analyse der Bestrahlung eines Tumors beschäftigt sich mit der Einstellung der Reichweite der α -Teilchen. Ziel der Analyse ist es, die Energie der α -Teilchen so einzustellen, dass die Energiedeposition im Tumor maximiert wird. Die Variation der Energie erfolgt dabei in einem Bereich von 200 MeV bis 400 MeV. Die Abbildung 5.9 zeigt den Anteil der im Tumor deponierten Energie von der Gesamtenergie der Teilchen. Teilchen der Energie von 200 MeV bis 280 MeV besitzen eine Reichweite, die kürzer als der Abstand des Tumors zur Schädeldecke ist. Daher erreichen sie den Tumor nicht und die

Energiedeposition verschwindet für diese Energien. Erreichen die Teilchen den Tumor, so nimmt die Energiedeposition stark zu und erreicht ihr Maximum bei einer Energie von 320 MeV. Eines dieser simulierten Ereignisse ist in Abbildung 5.8 dargestellt. Die maximale Energiedeposition wird erreicht, wenn sich der Bragg-Peak am hinteren Ende des Tumors befindet. Nach Erreichen des Maximums nimmt die Energiedeposition im Tumor wieder ab, da sich der Bragg-Peak hinter dem Tumor befindet. Die Energiedeposition verschwindet nach Erreichen des Maximums nicht wieder, sondern nimmt mit wachsender Energie einen konstanten Wert an. Die Energiedeposition im Tumor ist dann durch den Plateaubereich des Profils gegeben.

6. Zusammenfassung und Ausblick

Die Entwicklung des *Python*-Pakets `geant4_10_3_package` hat die Verwendung in den Praktika zu den Vorlesungen “Teilchenphysik I“ und “Detektoren für Teilchen- und Astroteilchenphysik“ zum Ziel. Wichtige Elemente der Konzeption des Pakets sind die Abstraktion und Vereinfachung der Verwendung von Geant4 sowie der möglichst nahe Ablauf an einer Geant4-Anwendung. Die Abstraktion wird durch die Verwendung von Methoden der `ApplicationManager`-Klasse erreicht.

Das Paket enthält Module und Komponenten, die in jeder Anwendung genutzt werden, sowie austauschbare Komponenten, die eine flexible und anpassbare Anwendung ermöglichen. Zu den durchgehend genutzten Komponenten gehören die Visualisierung der Detektorgeometrie und der simulierten Ereignisse, die Aktionsklassen, eine Methode zur Initialisierung der Anwendung und die Generation primärer Teilchen und Vertices. Zusätzlich zu den bereits in `Geant4Py` verpackten Referenzphysiklisten stehen zwei erweiterte Physiklisten zur Verfügung. Diese gehören ebenso zu den austauschbaren Komponenten, die eine Anpassung der Anwendung an die Simulation ermöglichen, wie die Definition verschiedener Geometrien. Verschiedene Musterlösungen der Beispielsimulationen demonstrieren die Verwendung der zur Verfügung gestellten Methoden zur Analyse der simulierten Ereignisse. Die Austauschbarkeit der Geometriedefinitionen bietet die Möglichkeit der Definition eigener Geometrien durch die Studierenden. Um dies zu unterstützen steht eine Beispieldefinition zur Verfügung, die die Beschreibung einer Geometrie demonstriert.

Während der Entwicklung des Pakets wurde besonderen Wert auf die Produktion gut strukturierten und dokumentierten Codes, gelegt. Das Layout des Codes orientiert sich daher an den Vorschlägen, die in der Anleitung für *Python*-Code, PEP 8[vR01], beschrieben werden. Die Dokumentation der Schnittstellen erfolgt über Docstrings, *Python*-interne Objekte, die der Dokumentation des Codes dienen, und die Software Sphinx[sph]. Sie befindet sich im Anhang der Arbeit.

Die erste Verwendung des Pakets erfolgt in den Übungen zur Vorlesung “Detektoren für Teilchen- und Astroteilchenphysik“, die am Ende des Wintersemesters 2016/2017 durchgeführt werden. Die Übungen führen zu einer ersten Rückmeldung der Studierenden. Auftretende Probleme und Verbesserungsvorschläge der Studierenden können somit bis zur Verwendung im Computerpraktikum zur Vorlesung “Teilchenphysik I“ im nächsten Wintersemester gelöst, respektive eingearbeitet werden. Weiterhin ist die Entwicklung einer graphischen Benutzeroberfläche angedacht. Diese dient der Erweiterung der bestehenden Visualisierungsmöglichkeiten um eine einfache Möglichkeit des Drehens der Ansicht und der Steuerung einfacher Simulationen über die graphische Oberfläche.

Literaturverzeichnis

- [Ago03] Agostinelli, S. et al: *Geant4-a simulation toolkit*. Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment, 506:250–303, 2003. doi:10.1016/S0168-9002(03)01368-8.
- [All06] Allison, J. et al: *Geant4 developments and applications*. IEEE Transactions on Nuclear Science, 53(1):270–278, 2006. doi:10.1109/TNS.2006.869826.
- [All12] Allison, J. et al: *The Geant4 Visualisation System - a multi-driver graphics system*. CoRR, abs/1212.6923, 2012. <http://arxiv.org/abs/1212.6923>.
- [All16] Allison, J. et al: *Recent Developments in GEANT4*. Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment, 835:186–225, 2016. doi:10.1016/j.nima.2016.06.125.
- [Ama81] Amaldi, U.: *Fluctuations in Calorimetry Measurements*. Physica Scripta, 23:409–424, 1981. doi:10.1088/0031-8949/23/4A/012.
- [Apo09] Apostolakis, J. et al: *Geometry and physics of the Geant4 toolkit for high and medium energy applications*. Radiation Physics and Chemistry, 78(10):859–873, 2009. doi:10.1016/j.radphyschem.2009.04.026.
- [Bic06] Bichsel, H.: *A method to improve tracking and particle identification in TPCs and silicon detectors*. Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment, 562:154–197, 2006. doi:10.1016/j.nima.2006.03.009.
- [Bru97] Brun, R. und Rademakers, F.: *ROOT - An Object Oriented Data Analysis Framework*. Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment, 389:81–86, 1997. Siehe auch: <http://root.cern.ch/>.
- [cma] *CMake*. <https://cmake.org/>.
- [Cou15] Coursey, J. S. et al: *Atomic Weights and Isotopic Compositions with Relative Atomic Masses*, 2015. [online]<http://physics.nist.gov/Comp/>. National Institute of Standards and Technology, Gaithersburg, MD. publiziert als: Meija, J. et al: *Atomic weights of the elements 2013. Pure and Applied Chemistry*, 88:265-291, 2016; Berglund, M. und Wieser, M. E.: *Isotopic compositions of the elements 2009 (IUPAC Technical Report)*. Pure and Applied Chemistry, 83(2):397-410, 2011. doi:10.1351/PAC-REP-10-06-02; Audi, G. et al: *The AME2012 atomic mass evaluation (I). Evaluation of input data, adjustment procedures*. Chinese Physics C, 36(12):1287-1602, 2012.
- [Fab03] Fabjan, C. W. und Gianotti, F.: *Calorimetry for particle physics*. Reviews of Modern Physics, 75(4):1243–1286, 2003. doi:10.1103/RevModPhys.75.1243.

- [Gab94] Gabriel, T. A. et al: *Energy dependance of hadronic activity*. Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment, 338(2-3):336–347, 1994. doi:10.1016/0168-9002(94)91317-X.
- [Gea16a] Geant4 Collaboration: *Geant4 User's Guide For Application Developers*, geant4 10.3 Auflage, 2016.
- [Gea16b] Geant4 Collaboration: *Physics Reference Manual*, geant4 10.3 Auflage, 2016.
- [Hei54] Heitler, Walter: *The quantum theory of radiation*. The international series of monographs on physics. Clarendon Pr., Oxford, 3. ed. Auflage, 1954.
- [Hun07] Hunter, J. D.: *Matplotlib: A 2D graphics environment*. Computing In Science & Engineering, 9(3):90–95, 2007. doi:10.1109/MCSE.2007.55.
- [Kol16] Kolanoski, Hermann: *Teilchendetektoren : Grundlagen und Anwendungen*. Springer Spektrum, Berlin, Heidelberg, 1. Auflage, 2016. <http://swbplus.bsz-bw.de/bsz468154302cov.htm><http://dx.doi.org/10.1007/978-3-662-45350-6>.
- [Oli14] Olive, K. A. et al. (Particle Data Group): *2014 Review of Particle Physics*. Chinese Physics C, 38(0100009), 2014. updated: 2015.
- [Pat16] Patrignani, C. et al(Particle Data Group): *2016 Review of Particle Physics*. Chinese Physics C, 40(100001), 2016.
- [Pyt] *python*. <https://www.python.org>.
- [Qua] Quast, G.: *Virtuelle Maschine mit Software zur Datenauswertung*. <http://www-ekp.physik.uni-karlsruhe.de/~quast/>.
- [Ros52] Rossi, Bruno: *High-energy particles*. Prentice-Hall physics series. Prentice-Hall, New York, 1952.
- [Säv13] Sävöiu, D.: *kafe-Ein Python-Paket für elementare Datenanalyse im Physikpraktikum*, 2013. www-ekp.physik.uni-karlsruhe.de/~quast/kafe/DSavoiu_kafe.pdf.
- [sph] *Sphinx-Python Documentation Center*. <http://www.sphinx-doc.org/en/1.5.1/index.html>.
- [Tsa74] Tsai, Yung Su: *Pair Production and Bremsstrahlung of Charged Leptons*. Reviews of Modern Physics, 46:815, 974. doi:10.1103/RevModPhys.46.815, 10.1103/RevModPhys.49.421.[Erratum: Rev. Mod. Phys.49,521(1977)].
- [TVi] *TVirtualMC Class Reference*. <https://root.cern.ch/doc/master/classTVirtualMC.html>.
- [vR01] Rossum, G. et al van: *PEP 8-Style Guide for Python Code*, 2001. <https://www.python.org/dev/peps/pep-0008/>.
- [Wig00] Wigmans, Richard: *Calorimetry : Energy Measurement in Particle Physics*. International series of monographs on physics ; 107. Clarendon Press, Oxford [u.a.], 1. publ. Auflage, 2000.

Anhang

A. Übungsblatt

Das von Wozniewski, S. und Zäh, N. erstellte Aufgabenblatt für die Übungen zur Vorlesung “Detektoren für Teilchen- und Astroteilchenphysik“ ist unter der Adresse

<http://www.ekp.kit.edu/~quast/detektorsimulation/Uebung16/>

zu finden.

B. Dokumentation

Abbildung B.1 zeigt die Titelseite der Dokumentation der Schnittstellen des Pakets `geant4_10_3_package`. Die vollständige Dokumentation ist unter

<http://www.ekp.kit.edu/~quast/detektorsimulation/>

einzusehen. Sie liegt in englischer Sprache vor.

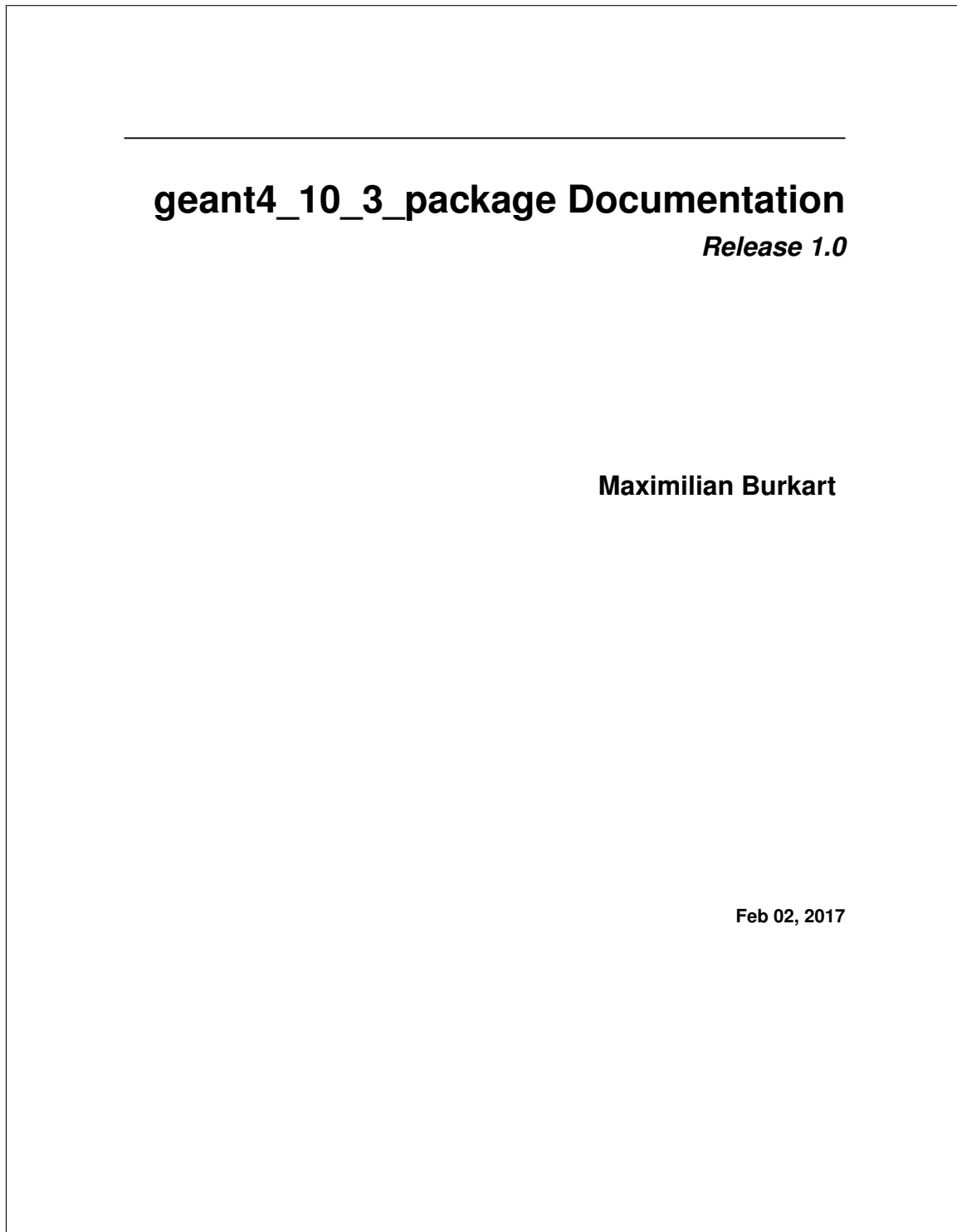


Abbildung B.1.: Titelseite der Dokumentation