

Spezielle Relativität

Jonas Berggren

February 25, 2020

Contents

1	Relativität nach Newtonscher Physik	3
1.1	Bezugssystem	3
1.2	Wechsel von Bezugssystemen	4
1.3	Minkoswsky Raumzeit diargamme	4
2	spezielle Relativität	4
2.1	Herleitung	5
2.2	transformation zwischen Bezubssytemen	5
2.3	Raumzeititnervall als erhaltene Größe	5
2.4	Implikationen	5
2.5	Einschränkungen dieses Modells	5
3	Programm	5
3.1	Nutzung	5
3.2	Button	6
3.2.1	Methoden	6
3.3	Input	7
3.3.1	Methoden	7
3.4	Box	7
3.4.1	Methoden	7
3.5	Obj	7
3.5.1	Methoden	7
3.6	Funktionen	7

4	Quellen	7
5	Notizen	7

Abstract

In diesem Dokument erkläre ich wie ich ein Programm entwickelt habe, was Albert Einsteins spezielle Relativität visualisiert. Das Programm arbeitet anhand von Minkowski-Raumzeitdiagrammen, und nutzt Objektorientierte Programmierung.

Außerdem erkläre ich die darunter liegende Physik und leite Lorentztransformation her. Dabei werde ich auf Relativität vorr Einstein, auf Minkowski-Raumzeit-diagramme, auf Transformationen zwischen Bezugssystemen, Auf die Implikation Spezieller Relativität, und desse Einschränkungen.

1 Relativität nach Newtonscher Physik

Bevor wir über Einsteins spezielle Relativität reden können, müssen wir das Konzept von Raum, Zeit und bewegung klarstellen

1.1 Bezugssystem

Zunächst muss klar gestellt werden wie Position, Zeit und Geschwindigkeit gemessen werden. Dazu muss ein Koordinatensystem Räumlich und Zeitlich definiert werden. Das Koordinatensystem hat einen Ursprung mit $x = 0, y = 0, z = 0$ und $t = 0$. Hierbei ist der Ursprung des Koordinatensystems in der Regel auf ein Objekt zu Beginn des Beobachtungszeitraums bezogen. Für diese Betrachtung müssen folgende Bedingungen erfüllt sein:

- Das Bezugssystem muss inertial(unbeschleunigt) sein
- Die Raumzeit muss flach sein, es darf keine Gravitation wirken, ART

Durch die Tatsache, dass in allen Inertialen Bezugssystemen die gleichen physikalischen Gesetze gelten, sind alle Bezugssysteme gleich gültig. Es ist keine Universal gültige Aussage über die Position oder Geschwindigkeit eines Körpers, oder Zeitpunkt eines Ereignisses möglich. Demnach ist es nichts sagend zu sagen, man hätte zum Zeitpunkt t die Position x, y, z und bewege sich mit Geschwindigkeit \vec{v} . Es muss immer ein Bezugspunkt gewählt werden z.b. Erdmittelpunkt Bezugssysteme können sich also relativ zu einander bewegen und dennoch gleichermaßen gültig das selbe Ereignis beschreiben.

1.2 Wechsel von Bezugssystemen

Ich werde mich im folgenden auf eine Raumdimension beschränke. Das Hinzufügen der anderen Raumdimensionen, kann Durch ersetzen der Richtungsabhängigen Größe n , durch Vektoren. x wird demnach zu $\vec{0p}$, v zu \vec{v} . dabei ist zu beachten das die gerichteten Relativistischen Effekte nur entlang der Bewegungsrichtung auftreten.

Es wird zunächst ein Bezugssystem gewählt mit den Größen x, t und v . Anschließend wird ein gestrichenes Bezugssystem gewählt mit den Größen x', t' und v' , wobei $t = t'$ gilt. Aus unsere alltäglichen erfahrung geht hervor, dass für die Position eines, zu dem ungestrichenen System statischen Objekt gilt: $x' = x - vt$ Genau so gilt für geschwindigkeiten:

$$u' = u - v \quad (1)$$

Hierbei ist v die Geschwindigkeit des Gestrichenen Bezugssystem und u Die geschwindigkeit des betrachteten objekts.

1.3 Minkoswsky Raumzeit diargamme

Das Minkowski Raumzeit Diagramm betrachtet, in seiner üblichen Form, Objekte in einer Raumdimension und Zeit. Hierzu wird die Zeit auf die vertikale Achse gelegt und die Position auf die horizontale. Für die Betrachtung von spezieller Relativität werden die Einheiten einfachheitshalber so gewählt, dass die Lichtgeschwindigkeit $c = 1$ und $x = t$.

2 spezielle Relativität

Die Spezielle Relativität fügt ein entscheidendes Postulat hinzu:

- Die Lichtgeschwindigkeit ist eine Universelle Konstante

Dies wirft direkt eine Frage auf: Wenn jemand mich mit einer Taschenlampe anleuchtet während ich mich auf ihn zu bewege, wie kann es dann sein, dass wir beiden den exakt gleichen wert für die Beschwindigkeit diese licht messen? Das Postulat ist also nicht mit der Formel 1 vereinbar.

2.1 Herleitung

Nehmen wir folgendes Szenario an:

Es wird eine Person auf der Erde und eine Person an Bord einer Rakete betrachtet, die sich relativ zur Erde mit einer Geschwindigkeit v bewegt. Hierbei ist das Bezugssystem des Astronauten gestrichen. An Bord der Rakete befindet sich eine Uhr, die Zeit misst, indem sie ein Photon, über eine Strecke l gegen einen Spiegel sendet und wartet, bis das Photon wiederkommt. Der Einfachheit halber bewegt sich das Licht dabei orthogonal zur Bewegungsrichtung der Rakete. Aus dem gestrichenen Bezugssystem ist die Zeit, die das Licht braucht $\Delta t' = \frac{2l}{c}$. Das Licht aus dem ungestrichenen Bezugssystem legt jedoch eine längere Strecke zurück, nämlich $s = \sqrt{l^2 + (v\Delta t')^2}$. Da die Lichtgeschwindigkeit in beiden Systemen identisch sein muss, muss $\Delta t' < \Delta t$ gelten.

2.2 transformation zwischen Bezugssystemen

2.3 Raumzeitintervall als erhaltene Größe

2.4 Implikationen

2.5 Einschränkungen dieses Modells

3 Programm

Das Programm basiert auf objektorientierter Programmierung und nutzt das pygame Modul für die Grafik. Ich habe insgesamt vier Klassen entwickelt: Eine für die Darstellung der Bezugssysteme im Diagramm, eine für die Darstellung von Buttons und zwei für die Darstellung der Eingabefelder. Außerdem habe ich jeweils eine Funktion zum Wechseln von Bezugssystem, und zum Handhaben von unzulässigen Nutzereingaben.

3.1 Nutzung

Dieses Programm in Python 2.7 ausgeführt werden. In dem rechten Feld können Geschwindigkeit und Startposition von Objekten eingegeben werden. Beim Klicken der Send-Taste wird die Weltlinie, und die dazu gehörige Gleichzeitigkeitslinie des Objektes in das Koordinatensystem eingezeichnet. Es können

nach beliebigen Objekten hinzugefügt oder entfernt werden. Alle sichtbaren Weltlinien können angeklickt werden um die Lorentz-Transformation in das jeweilige Bezugssystem beobachten zu können.

3.2 Button

Die Klasse `Button` dient zur Darstellung und Handhabung aller Buttons. Diese Klasse nimmt einen Typ und optional, einen parent als Argument.

3.2.1 Methoden

In dem Konstruktor wird die Box in Abhängigkeit von Typ definiert. Anschließend werden weitere Instanzvariablen, für Farbe, Aktivität, Text, Typ, Mutterobjekt definiert.

Die Methode `handle` wird in der Hauptschleife aufgerufen und nimmt pygame Ereignisse als Argumente. In Abhängigkeit vom Typ wird der Ortsvektor der Maus angepasst. Wenn sich die Maus über ein dem Button befindet wird der Aktivitätszustand auf True gesetzt, andernfalls auf False. Je nach Aktivitätszustand wird in der Methode `draw` die Farbe geändert, was ein Hoverfunktion darstellt. Befindet sich die Maus über `self`, wird dann bei einem Klick, zwischen den Typen unterschieden. Ist `self.type` gleich `add`, wird der Liste der `objs` eine Instanz von `Obj` hinzugefügt. Außerdem wird die Position von `self.rect` angepasst.

Ist `self.type` gleich `send` wird der Index von `self` ermittelt, und der Methode `enter` übergeben, die in Abschnitt 3.3.1 näher erläutert wird.

Ist `self.type` gleich `ok` wird die globale Variable `err` None gesetzt.

Ist `self.type` gleich `x` wird der Index von `self` ermittelt. Anschließend wird dieser Index genutzt, um die jeweiligen Einträge in den Listen `objs`, `delbuttons`, `inputs` und `sendbuttons`. anschließend werden die Position von den verbleibenden Elementen angepasst um das gelöschte Element aufzufüllen.

Die Methode `draw` wird in der Hauptschleife aufgerufen und zeichnet je nach Typ `self.rect` und `self.txt` auf die unterschiedlichen Flächen.

3.3 Input

3.3.1 Methoden

3.4 Box

3.4.1 Methoden

3.5 Obj

3.5.1 Methoden

3.6 Funktionen

Die Funktion error wird von der Funktion enter bei jedem Fehlerfall aufgerufen. Es wird ein fallspezifischer Fehlerwert übergeben. Die Funktion error erstellt ein Fenster auf dem eine fehlerspezifische Nachricht angezeigt wird. Außerdem wird ein 'ok' button erstellt mit dem das Fenster wieder geschlossen werden kann.

4 Quellen

Loedel-Minkowski-Diagramm; zweidimensionale Raumzeit (Zugriff: 06.09.2019):

<https://stackoverflow.com/questions/46390231/how-to-create-a-text-input-box-with-pygame>

<https://www.youtube.com/playlist?list=PLD9DDFBDC338226CA>

5 Notizen

$x = ct$ $x' = ct - vt = (c-v)t'$ speed addition Raum, Zeit, Gleichzeitigkeit, Relativität Invariante Proper time $\Delta t = 0$ they can be connected by a lightbeam spacelike, timelike lightcone Fourvector Fourvelocity Gleichzeitigkeit $t = vx$ Linie der Gleichzeitigkeit ist die Spiegelung der Weltlinie entlang der Lichtlinie