

Form

- ☐ Erstens solltet ihr solche Dokumente ausschließlich als PDF versenden, da je nach Version und Programm zum Öffnen von docx Dateien, die Formatierung anders aussehen kann.
- ☐ Zweitens habt ihr hier extrem viele Inkonsistenzen, was Leerzeilen vor und nach Überschriften angeht.
- ☐ Drittens habt ihr, soweit ich es sehen kann, 4 verschiedene Stufen von Überschriften, die ihr nicht konsistent einsetzt.
- ☐ "Runtime and development environment" ist eine Stufe 3 Überschrift, genau wie "Euclidean reflection", die Darstellung ist aber eine komplett andere. Das würde noch stärker auffallen, wenn ihr, wie üblich und von uns gewünscht, die Kapitel Nummern anzeigen lasst, sowie ein Inhaltsverzeichnis erstellt.
- ☐ Viertens sind eure Referenzen keine Referenzen, sondern plain Text.
- ☐ Es fehlen Referenzen bei Bildern, ihr habt nur zwei Referenzen überhaupt verwendet und beides mal bei übermäßig langen Zitaten, die ihr besser in eurem Worten umschreiben könntet.

- ☐ Euer Referenzverzeichnis hat auch nicht die Richtige Form. Bitte seht euch Reference Styles an und sucht euch einen davon aus. Paper werden nicht über den Link zu einer PDF referenziert, sondern über Titel, Author und Journal etc. Webseiten werden mit dem "offiziellen" Namen, Author (wenn möglich) und Zugriffsdatum etc. referenziert. Eure Gleichungen sind nicht als Gleichungen zu erkennen, da sie fließend im Text stehen. Das gleiche gilt für Code.
- ☐ Es fehlt auch noch ein komplettes Deckblatt, mit Betreuer, Datum, Fortgeschrittenen Praktikum etc.
- ☐ Bitte verwendet einfach das Latex Template, dann passen die Überschriften automatisch, die Form ist konsistent, Quellen werden automatisch richtig angegeben, wenn ihr alle Informationen eintragt, Referenzen sind tatsächliche Referenzen und auch Gleichungen und Code lassen sich sehr einfach als solche kennzeichnen.

Inhalt

- ☒ ~~Abstract ist zu kurz und ihr solltet ihn einmal auf deutsch~~
- ☒ ~~und einmal auf englisch haben.~~
- ☒ ~~Ich würde in der Introduction nicht direkt mit einem Zitat anfangen. Generell sind wörtliche Zitate eher ungewöhnlich.~~
- ☒ ~~Wenn Figure 1 auch aus dem Paper kommt, müsst ihr auch der Figure eine Referenz geben.~~
- ☒ ~~Auch finde ich die Figure nicht gut, da in dem Beispiel Euklidische und Simplektische Reflektion (fast) identisch sind~~
- ☒ ~~Was genau soll "outer billiard reflection" sein?~~
- ☒ ~~Ich würde nicht "reflection algorithm" sagen, sondern eher sowas wie "reflection concept".~~
- ☒ ~~Außerdem würde ich bei Euclidean eher mit Optic reflection anfangen und dann sagen, dass es euklidisches Billiard ist, da es so für Leser besser zu verstehen ist~~
- ☒ ~~Ihr benutzt den Begriff "trajectory" schon bevor ihr ihn einführt. Entweder geht ihr davon aus, dass der Begriff so basic ist, dass ihr ihn nicht einführen müsst, oder ihr führt ihn ein, bevor ihr ihn nutzt. Dann würde ich mit Trajectory bevor, statt nach, Reflection gehen~~

- ☐ Eigentlich alles in der Introduction könnte eher in ein Fundamentals. In der Introduction solltet ihr eher erklären, was das Problem ist, das ihr versucht zu lösen und warum es ein Problem ist, das es Wert ist gelöst zu werden
- ☐ Eure Motivation würde ich aufteilen in zwei Bereiche, der tatsächlichen Motivation, die ihr für eure Introduction verwenden könnt, und der vielen Anwendungsbereiche, die ihr super für die related work nutzen könnt.
- ☐ Die Related Work sollte natürlich auf die Paper hinweisen auf die ihr euch massiv bezieht, sie soll eure Arbeit aber auch in Kontext zu anderen Arbeiten setzen. Ihr habt in der Motivation viele Anwendungsgebiete dafür genannt. Hier könnt ihr dann erklären, was eure Arbeit in dem Bereich für einen Vorteil bieten würde.
- ☐ Achtet hier bitte besonders auf die Zitationsweise. Es werden nie alle Autoren ausgeschreiben, im Normalfall der erste Author + "et al.". Auch der Paper Name muss nicht unbedingt genannt werden, wenn dies für die Differenzierung nicht notwendig ist. Und auch hier sind direkte Zitate eher unüblich.
- ☐ Nach der Related Work könnt ihr ein Kapitel "Fundamentals" einfügen, wenn es Grundlagen gibt, die für eure Arbeit notwendig sind, keinem speziellen Paper zugeordnet sind, und nicht von euch erarbeitet wurden. Dazu gehören zum Beispiel all die Mathematischen Grundlagen zur Reflektion, die ihr in euren Quellen habt.
- ☒ ~~Danach kommt ein Kapitel, was typischerweise "Method" genannt wird, was eurem "Our Approach" entspricht.~~
- ☐ Was ihr unter "Overview" beschreibt, ist eher Teil der Result Section, da es hier um die konkrete Anwendung geht. Insbesondere nutzt ihr hier viele Begriffe und Eigenschaften eurer Anwendung, die ihr noch nicht eingeführt habt.
- ☐ Hier sollte eher etwas allgemeines hin, im Sinne von "Wir haben für die Analyse vom Billiard eine Anwendung erstellt. Die basiert auf folgenden Reflektionsmodellen und Eigenschaften. Anschließend erläutern wir die konkrete Implementierung"
Die Reflection Methods sind eigentlich der Kernteil eurer Arbeit, der sollte deutlich ausführlicher sein. Hier könnt ihr ruhig etwas weiter vorne anfangen mit Teilchen die sich auf einem Pfad bewegen, bis sie durch ein Objekt abgelenkt werden und dann die verschiedenen Reflection Models erklären. Auch die ruhig deutlich ausführlicher. Warum ist euklidische Reflektion so wichtig? Wieso ist diese so wie sie ist? Wo findet sie alles Anwendung? Wieso ist sie für eure Arbeit wichtig?
- ☒ ~~Was ist "reflectionLineSymplectic"?~~
- ☐ Warum habt ihr die symplectic reflection nicht auch für nicht-konvexe Polygone gemacht? Was hätte dafür ergänzt werden müssen?
- ☐ Runtime and ...: Was wird genutzt? Wofür wird es genutzt? Warum wird es genutzt? Was sind die Vor- und Nachteile davon? Ich würde den Teil etwas umstrukturieren, damit die Fragen etwas getrennter beantwortet werden, aktuell werden insbesondere die ersten beiden etwas durcheinander gewürfelt. Ich würde auch die Aufzählung raus nehmen und es in Fließtext umwandeln
- ☒ ~~Code Structure: Hilfsklassen wie "Trajectories" würde ich vermutlich gar nicht nennen.~~
- ☐ Was die exakte Struktur angeht müsst ihr nicht so sehr ins Detail gehen, wichtiger ist die konzeptionelle Struktur
- ☐ Results: Fehler und Mängel, die einem Leser nicht automatisch klar werden, braucht ihr so explizit nicht zu adressieren. (Also eigentlich natürlich schon, aber das weiß ja dann keiner)
- ☐ Seid ihr euch sicher, dass es im Euklidean Mode keine periodischen Zyklen gibt? Intuitiv kommt mir das nämlich komisch vor. Falls nicht müsst ihr einem Leser klar machen, warum das nicht passiert. Und ihr müsst Kritik entschärfen können. Ist es vielleicht nur eine Ungenauigkeit in eurer Berechnung, die dazu führt, dass keine Zyklen entstehen?

- ☐ Figure 11/12: Beide Beispiele starten am gleichen Punkt in gleichem Abstand zu 90 Grad, also entsprechend als wären sie reflektiert, einer vorwärts, einer rückwärts. Würde es eine Periodizität geben, müssten entsprechend beide exakt das gleiche anzeigen. Eure Argumentation ist da etwas komisch. Auch hier bin ich mir bei dem ausbrechen nicht sicher, ob das nicht einfach ein Rundungsfehler ist
- ☐ Habt ihr für eure 3 Behauptungen am Ende der Results bewiese? Wie erklärt ihr euch dieses Verhalten? Was kann man mit diesen Informationen anfangen?
- ☐ Bitte beschreibt eure Grafiken nicht nur, sondern erklärt, was man da sieht und warum man es sieht und was das bedeutet.

Title: Visualization of Symplectic and Euclidean Billiards in Polygons: A JavaScript-based Web Application

Abstract:

This paper presents a JavaScript-based web application that enables the simulation and visualization of Symplectic and Euclidean billiards within two-dimensional convex and non-convex polygons. Billiards, a classical problem in mathematics, involve the motion of a point particle within a closed polygonal boundary, reflecting off its edges according to specific rules. By implementing an intuitive and interactive user interface, our web application allows users to explore the fascinating dynamics of billiards in various polygonal shapes.

The application leverages modern web technologies to provide a seamless and immersive experience. Through the utilization of the HTML5 canvas element and the power of JavaScript, we enable real-time rendering and dynamic interactions between the particle and the polygon. Users can specify the initial conditions, such as the particle's position, velocity, and the polygon's shape, and observe the subsequent trajectory of the particle as it undergoes reflections and collisions with the polygon's boundaries.

The visualization of both Symplectic and Euclidean billiards is a unique feature of our web application. Symplectic billiards, characterized by preserving the phase space volume, offer intriguing dynamics and have applications in various areas of physics and mathematics. On the other hand, Euclidean billiards adhere to the classical rules of reflection, providing a more intuitive understanding of billiard dynamics. Our application allows users to switch between these two modes, enhancing the versatility and educational value of the simulation.

We demonstrate the capabilities of our web application through various examples, including billiards within convex and non-convex polygons of different complexities.

Additionally, we provide tools for visualizing and analyzing the properties of billiard trajectories, such as energy conservation, invariant curves, and stability regions. These features enable users to gain insights into the behavior of billiards in different polygonal environments and explore the rich mathematical structure underlying these systems.

In conclusion, our JavaScript-based web application offers an accessible platform for simulating and visualizing Symplectic and Euclidean billiards in two-dimensional convex and non-convex polygons. By providing an intuitive interface, real-time rendering, and a range of analytical tools, this application opens up new avenues for studying the dynamics of billiards and facilitates exploration and experimentation in the field of mathematical physics.

Keywords: Symplectic billiards, Euclidean billiards, two-dimensional polygons, JavaScript, web application, visualization, mathematical physics.

Abstract:

This report presents a JavaScript-based web application that simulates and visualizes Symplectic and Euclidean billiards within two-dimensional convex and non-convex polygons. It explores various interesting cases in both types of billiards and provides a comparative analysis between them. The report includes a detailed explanation of Symplectic and Euclidean billiards, along with a methodology section describing the algorithms and techniques used in the web application. The user interface and features of the application are discussed, highlighting its interactive controls and unique functionalities. Results from the simulations and visualizations are presented, with an emphasis on specific examples and their outcomes. A comprehensive comparison between Symplectic and Euclidean billiards is provided, examining their similarities, differences, and practical implications. The report concludes with a discussion on the significance of the findings and potential future research directions. Overall, the web application serves as a valuable tool for exploring and understanding the dynamics of billiards in different polygonal environments.

