­

Biljard

Inlämningsuppgift i Spelfysik

Grundnivå 3 högskolepoäng

Höstterminen 2014

Sebastian Zander

Innehållsförteckning

[1 Fysikaliskt fenomen 1](#_Toc408066944)

[1.1 Stöt 1](#_Toc408066945)

[1.2 Glidande och rullande kula 2](#_Toc408066946)

[1.3 Glidande kula 2](#_Toc408066947)

[1.4 Rullande kula 3](#_Toc408066948)

[2 Simuleringsmodell 5](#_Toc408066949)

[3 Implementation 7](#_Toc408066950)

[4 Utvärdering 8](#_Toc408066951)

[4.1 Tabeller 8](#_Toc408066952)

[5 Diskussion och slutsats 9](#_Toc408066953)

[5.1 Förbättringsförslag 9](#_Toc408066954)

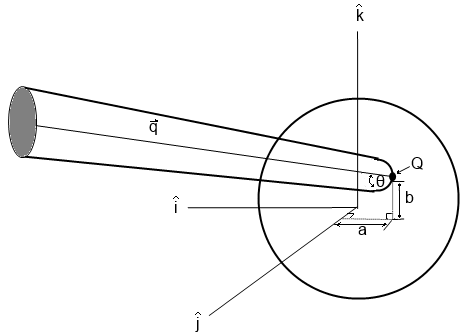
[Referenser 10](#_Toc408066955)

# Fysikaliskt fenomen

Det fysikaliska fenomen som tas upp i denna rapport är hur en stillastående biljardkula efter stöt beter sig.

## Stöt

En billjardkula kan sättas i rullning från en stöt av en billjardkö (även vid kollision från annan kula, som ej behandlas här). Stöten kan beskrivas med parametrarna *a*, *b* och *θ* som viasas i Figur 1 samt *V0*, vilket är köns hastighet direkt vid stöd (Leckie & Greenspan, 2006).



Figur 1

Vinkeln *θ* beskriver köns vinkel mot det horisontella planet (-). Punkten *Q* ligger även på ett avstånd *c* från kulans mittpunkt längs -axeln, där . Således är alltså .

Utgångshastigheten beräknas utifrån dessa värden. Newton’s andra lag integreras för att ge kulans hastighet efter köslaget. Kollisionstiden under stöttiden antags vara försvinnande kort, vilket förenklar beräkningarna men visar sig även vara empiriskt korrekt i ordningen 200 mikrosekunder i köslag på 1 m/s enligt Marlow (1995), refererad i Leckie & Greenspan (p. 216). Givet ett sådant antagande kan kraften behandlas som en elastisk impuls och integrationenen av Newton’s andra lag över en kort tid ger att (Leckie & Greenspan, 2006).

Kulans utgångshastighet ges då som:

|  |  |  |
| --- | --- | --- |
|  |  | (1) |

Kraften F fås enligt Leckie, W & Greenspan, M (2006) av:

|  |  |  |
| --- | --- | --- |
|  |  | (2) |

då R är kulans radius, m kulans massa och M köns massa. Vinkelhastigheten fås med liknande resonemang som hastigheten:

|  |  |  |
| --- | --- | --- |
|  |  | (3) |

där tröghetsmomentet för kulan.

## Glidande och rullande kula

Direkt efter stöt befinns sig kulan i ett glidande läge. Kulan har vid varje givet tillfälle en hastiget och en vinkelhastighet. Dessa två hastigheter kan vara riktade i olika riktningar. Allteftersom att kulan glider över bordet ändras dessa hastigheter beroende på friktionen mellan kulan och bordet. Vid en viss tid kommer de två hastigheterna att match varandra () och kulan övergår till ett rullande läge och friktionen mellan kulan och planet försvinner nästan helt (Cross, 2012). är friktionskoefficiententen för den glidande kulan, vilken generellt är skilt från friktionskoefficienten vid rullning, . Typiska värden i billjard för 0.15 – 0.4 är generellt ungefär och värden för är ungefär 0.005 – 0.015 (Alciatore, u.d.).

Marlow, enligt Leckie & Greenspan, 2006, (p. 217), använder en relativ hastighet vid den punkt där kulan vidrör bordet för att beskriva vilket läge kulan befinner sig i:

|  |  |  |
| --- | --- | --- |
|  |  | (4) |

Kulan befinner sig i glidande läge när och i rullande läga när .

## Glidande kula

Enligt Leckie & Greenspan, (2006) ges kulans position , hastighet och vinkelhastigheten i glidande läge av formlerna:

|  |  |  |
| --- | --- | --- |
|  |  | (5) |

|  |  |  |
| --- | --- | --- |
|  |  | (6) |

|  |  |  |
| --- | --- | --- |
|  |  | (7) |

Den relativa hastigheten kan enligt formlerna ovan även beskriva som:

|  |  |  |
| --- | --- | --- |
|  |  | (8) |

kan beräknas med föregående formel för . Ekvationerna är uttryckta i kulans referensram, därav subskriptet *B*, så att -axeln är riktad mot kulans rörelseriktning. För att finna positionen för kulan i bordets referensram kan en rotationsmatris appliceras:

|  |  |  |
| --- | --- | --- |
|  |  | (9) |

Tiden för det glidande stadiet av kulan ges av funktionen:

|  |  |  |
| --- | --- | --- |
|  |  | (10) |

## Rullande kula

Den rullande kulans läge beskrivs av, enligt Leckie & Greenspan, (2006):

|  |  |  |
| --- | --- | --- |
|  |  | (11) |

|  |  |  |
| --- | --- | --- |
|  |  | (12) |

|  |  |  |
| --- | --- | --- |
|  |  | (13) |

# Simuleringsmodell

Simuleringen ignorerar rörelse i -axeln och kulan rör sig således endast i --planet. Därav kan simuleringen göras helt i 2D, med undantag av stöt-parametrarna.

Simuleringen använder euler-integrering genom derivering av , så att och (Leckie & Greenspan, 2006). Den enda kraften som agerar på kulan, och därmed accelerar kulan () är friktionskraften. Friktionskraften ges av:

|  |  |  |
| --- | --- | --- |
|  |  | (14) |

Den relativa hastigheten visar rörelsen för den spinnande kulans kontaktpunkt på bordet. Friktionskraften sker i motsatt riktning till denna hastighet:

|  |  |  |
| --- | --- | --- |
|  |  | (15) |

Med Newton’s andra lag ger detta förändringen i hastighet med tiden:

|  |  |  |
| --- | --- | --- |
|  |  | (16) |

|  |  |  |
| --- | --- | --- |
|  |  | (17) |

|  |  |  |
| --- | --- | --- |
|  |  | (18) |

Vinkelhastighetens förändring med tiden går på liknande vis att få fram med hjälp av friktionsvridmomentet, :

|  |  |  |
| --- | --- | --- |
|  |  | (19) |

|  |  |  |
| --- | --- | --- |
|  |  | (20) |

|  |  |  |
| --- | --- | --- |
|  |  | (21) |

|  |  |  |
| --- | --- | --- |
|  |  | (22) |
|  |  |  |
|  |  | (23) |

Där .

Formell (8) antyder att den relativa hastigheten aldrig ändrar riktning. Eftersom att endast riktningen används för att uppdatera vinkelhastigheten och hastigheten så skulle det räcka med att räkna ut endast en gång. Magnituden är dock viktig för att ta reda på när kulan börjar rulla. kan då användas och fås, från formell (8), till:

|  |  |  |
| --- | --- | --- |
|  |  | (24) |

blir då irrelevant att räkna på under uppdatering av kulan sålänge simuleringen inte kommer simulera spinn visuellt. Detta kan även ge fördelar rent beräkningsmässigt. Kulan börjar rulla så snart , eftersom att någon gång mellan t och t + . På detta vis kan simuleringen ändra kulans läge från spinnande och rullande, vilket påverkar vilken friktionskoefficient som ska användas. Under rullande läge är friktionskraftens riktning riktad i motsatt riktning till hastigheten (då ) så istället gäller, under detta stadie, för , men friktionskoefficienten för rullande kula:

|  |  |  |
| --- | --- | --- |
|  |  | (18) |

Värdena i simuleringen kan jämföras med de exakta värdena från formlerna i kapitel 1: Fysikaliskt fenomen, för att kontrollera korrekthet.

# Implementation

Simuleringen implementeras i spelmotorn Unity3D. Det finns två klasser som är viktiga: en för för att sköta användarinput och översätta de till stötparametrar (Main) och en för att sköta fysiken efter stöt (PhysicsModel). Main genererar stötparametrar från användarinput och skickar detta vidare till PhysicsModellsom då sätter kulan i rullning.

PhysicsModel befinner sig i tre lägen: Still, Sliding eller Rolling. När läget går tillbaka till Still (efter en stöt) återställs simuleringen av main och en ny stö kan göras. Simulerinen kan också återställas genom att trycka på mellanslag.

Två kulor används, en som simulerar kulan och en som använder exakta värden från formler. Detta görs för att kunna jämföra hur korrekt simuleringen är. Samma klass används för detta, bara genom att en bool sätts som gör att uppdateringsfunktionen antingen integrerar eller beräknar nästa position. Det finns inget skäl till att dela upp funktionen i två klasser.

Integrering görs med fasta tidssteg. I Unity finns en funktion för detta (FixedUpdate) men det går inte att ändra hur långt tidssteget är. Därför ackumuleras deltat för varje uppdatering. När tillräckligt högt delta har ackumulerats, alltså att deltat är lika med eller högre en det tänkta tidssteg, subtraheras tidssteget från deltat och PhysicsModel uppdateras tills det ackumulerade deltat återigen är mindre än tidssteget, och så vidare. På detta sätt fås ett fast tidssteg som går att ställa in.

Användaren styr parametrarna med musen. Genom att hålla inne höger musknapp flyttas a och b över kulan med musrörelsen. Om inte höger musknapp är nedtryckt styrs de horisontella och vertikala vinklarna istället. Genom att hålla ner vänster musknapp styr användaren hastigheten på stöten och genom att släppa vänster musknapp igen genomförs själva stöten.

# Utvärdering

Utan tillgång till ett biljardbord går det ändå att utvärdera simuleringen.

Ett sätt att utvärdera simuleringen är genom att jämföra resultaten för simulerinen med resultaten från de exakta värdena från formler beskrivna i kapitel 1. Detta ingår i implementationen och en grå kula visar de exakta värdena, medan en vit kula visar värdena från simuleringen, med integrering som beskrivet i kapitel 2. Kulorna följer ungefär samma kurva, men simuleringen halkar lite efter med tiden, vid ett tidssteg på 16 ms. Simuleringen rör sig alltså långsammare än den ska (enligt den exakta uträkningen). Om tidssteget sänks till 1 ms kommer simuleringen fortfarande att halka efter något, men inte lika mycket.

Ett annat sätt att utvärdera simuleringen är att jämföra med andra beskrivningar i andra artiklar. (Alciatore, 2005) (diagram 3) beskriver kulans rörelse över bordet vid en s.k. massé-stöt . Vid en massé-stöt hålls kön nästan vertikal. I exemplet stötes kön lite till vänster om kulans mitt. Simulering kan användas för att replikera en sådan stöt och kurvan kan jämföras. Simuleringen visar sig kunna replikera denna stöt och kulan kommer följa en sådan kurva. Diagrammet i artikeln visar inte avståndet, men på detta sätt kan det verifieras att simuleringen åtminstone kurvas åt rätt håll.

# Diskussion och slutsats

Kulan påverkas av flera krafter som beskrivs av icke-triviala formler och med olika referenssystem. Därför kan jag inte vara säker på om jag har förstått dessa formler rätt, och alltså kan jag inte vara säker på om kurvorna av kulan som använder exakta positioner är korrekt. Det är troligt att formlerna har missuppfattats, då kulan beter sig konstigt vid övergång från glidande till rullande (vid använding av exakta värden) under vissa förhållanden.

Dessutom så rör sig kulan i simuleringen helt fritt, utan exempelvis bordskanter. Eftersom att denna sitatuation blir väldigt olik en typisk situation i ett biljardbord är det svårt att se intuitivt om simulering beter sig korrekt enligt erfarenhet i verkligheten.

Jag har inte tillgång till ett biljardbord så jag har inte kunnat göra mina egna experiment och har därför varit tvungen att utvärdera simuleringen med andra metoder. Experiment som har gjorts fokuserar ofta på hur kulor beter sig vid krock, antingen med annan kula eller med bordskant. Återigen uppkommer ett liknande problem som beskrivs ovan, nämligen att experiment (de jag hittat) är begränsade till ett bord, och inte visar hur en kula beter sig över en längre distans utom via matematiska modeller.

## Förbättringsförslag

Från detta skulle man kunna gå vidare till kollisionshantering i simulatorn, både för andra kulor och kanter i bordet.

Referenser

Alciatore, D. G., 2005. Coriolis was brilliant ... but he didn’t have a high-speed camera – part I: Introduction. *Biliards Digest.*

Alciatore, D. G., u.d. *Physics of Pool and Billiards.* [Online]   
Available at: http://billiards.colostate.edu/threads/physics.html  
[Använd 06 01 2015].

Cross, R., 2012. Rolling motion of a ball spinning about a near vertical axis. *The Physics Teacher, 50,* pp. 25-27.

Leckie, W. & Greenspan, M., 2006. Pool Physics Simulation by Event Prediction. *ICGA Journal,* December.pp. 215-222.