Rörelsesimulering av ett sfäriskt objekt

Med programmering

Nadim Lakrouz

NA18

Birgittaskolan i Linköping

Handledare: Erik Skäringer

**Sammanfattning**

Innan man producerar något, utför olika forskare och företag miljontals olika simuleringar. Simuleringar förenklar och hjälper till att samla in olika data. Syfte med detta arbete är att studera fysik och genomföra olika rörelse simuleringar av ett sfärisk objekt. Vilka i framtiden kommer att kunna svara på sådana frågor som:

* Hur luftmotstånd påverkar rörelse av en objekt ?
* Hur Magnus kraften påverkar rörelse av en objekt ?

För att utföra simuleringar och få nödvändiga data används programmeringsspråket C# och Unity-spelmotorn. Den insamlade och studerade data visade förväntade resultat. Resultatet visade hur olika faktorer i luftmotstånd formeln påverkar objektets hastighet. Också hur linjär och vinkelhastighet i magnus kraft formel påverkar flygbanan. Av dessa resultat kan man dra slutsatsen att luftmotståndet påverkar flyghastigheten och magnus-kraften påverkar banan.

**Abstract**

Before producing anything, different researchers and companies perform millions of different simulations. Simulations simplify and help collect different data. Purpose of this work is to study physics and conduct various motion simulations of a spherical object. Wich in the future will be able to answer such questions as:

* How does air resistance affect the movement of an object ?
* How does Magnus force affect the movement of an object ?

To perform simulations and obtain the necessary data, the C# programming language and the Unity game engine are used. The collected and studied data showed expected results. The result showed how different factors in the air resistance formula affect the speed of the object. Also how linear and angular velocity in magnus force formula affects the flight path. From these results, it can be concluded that air resistance affects the flight speed, and the magnus force affects the trajectory.

**Innehåll**

[**1. Inledning**](#_pz0bbcqbqci2) **4**

[1.1 Syfte](#_twfq0yv9970k) 4

[1.2 Frågeställning](#_5s67a5vqkh27) 4

[1.3 Avgränsning](#_s51v01e2ia4d) 5

[1.4 Teori](#_ao3mhw9pemsb) 5

[**2 Material och metod**](#_oiai4gi72lh4) **8**

[2.1 Material](#_1ovnhcb2xgv5) 8

[2.2 Metod](#_vu6kymdh5r9) 8

[**3 Resultat och analys**](#_iiyzte52muhz) **10**

3.1 Luftmotstånd simulering 10

3.2 Magnus kraft simulering 11

[**4 Diskussion och slutsats**](#_q9tc2kd112oe) **13**

4.1 Fysiska krafter och dess verkan 13

4.1.1 Luftmotstånd 13

4.1.2 Magnus kraften 13

4.2 Egna slutsatser 14

4.2.1 För- och nackdelar med metod 14

4.2.2 Egna tankar 14

4.3 Slutsatser 15

[**4. Källförteckning**](#_17dp8vu) **15**

4.1 Otryckta källor 15

4.2 Tryckta källor 15

**5. Bilagor 16**

# 1. Inledning

I det här arbete kommer övervägas beteende och rörelse av sfäriska föremål under flygning. Med uppmärksamhet åt de krafter som verkar på objekt under detta fenomen.

Om vi betraktar en person som observatör, är processen att studera oftast närvarande i sportspel och studieområdet kan övervägas i olika sporter: från klassisk engelsk fotboll till kulstötning. Arbetet kommer att teoretiskt beskriva hela processen som en person ser i praktiken och förklara det matematiskt och sedan bygga en datorsimulering genom att använda spelmotorer för att genomföra objektets matematiska logik och fysiska beteende.

Den teoretiska delen av arbetet kommer att omarbetas specifikt för spelmotorn och programmeringsspråket. Programvaran byggd på spelmotorn gör det möjligt att inte bara överväga den matematiska komponenten utan också grafiskt se rörelsen av sfäriska föremål.

## 1.1 Syfte

Konstruktion och forskning av en matematisk modell. Som kommer att användas för att styra rörelsen hos ett sfäriskt objekt med hänsyn till olika fysiska faktorer.

## 1.2 Frågeställning

* Hur luftmotstånd påverkar rörelse av en objekt ?
* Hur Magnus kraften påverkar rörelse av en objekt ?

## 1.3 Avgränsning

Detta arbete kommer endast att hänvisa till teori och simulering. Således begränsar de naturliga testerna. Naturliga tester är svåra att producera och det kräver finansiella investeringar i utrustning för korrekta resultat. Modellen i denna simulering begränsar faktorer som vindriktning och vindhastighet. Temperaturen i miljön och i objektet. Objektets massa är statisk hela tiden och dess formen är sfärisk.

## 1.4 Teori

Grundlagen för mekanik är Newtons andra lag (Fysikguiden, 2021-03-14), som hänför sig till resultatet av yttre krafter som verkar på kroppen, dess massa och acceleration, till följd av krafternas påverkan. Kastets rörelse är inget annat än rörelse som uppstår på olika axlar. Boken beskriver ofta rörelsen av ett objekt i den 2: a dimensionen se (Alphonce m.fl. 2013, sidan 40). I detta fall sker rörelsen på 2 axlar. Hastigheten på olika axlar samt positionen kan beskrivas på följande sätt :

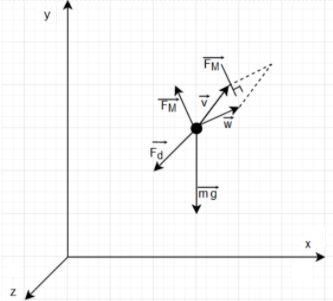
Hastighet i x- och y- led :



Läget i x- och y-led :

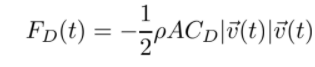


Tidigare beskrivs rörelse i 2-dimensionellt utrymme, men nu är det logiskt att överväga objektets beteende närmare den verkliga modellen, vi kommer att inkludera en extra axel för genomförandet av 3-dimensionellt utrymme. I detta fall är vektor variablerna, såsom hastigheten, redan beroende av den nya z-axeln.



I de tidigare ekvationerna och formlerna, för rörelse av ett objekt i 2-dimensionellt utrymme, försummade vi sådana faktorer som styr objektets rörelse med luftmotstånd, (Alphonce m.fl. 2013, sidan 60) och Magnuseffekten (Wikipedia, 2021-01-28), vi kommer att ta hänsyn till dessa faktorer i ekvationerna nedan för att komma ännu närmare de verkliga förutsättningarna för att studera beteendet hos ett sfäriskt objekt.

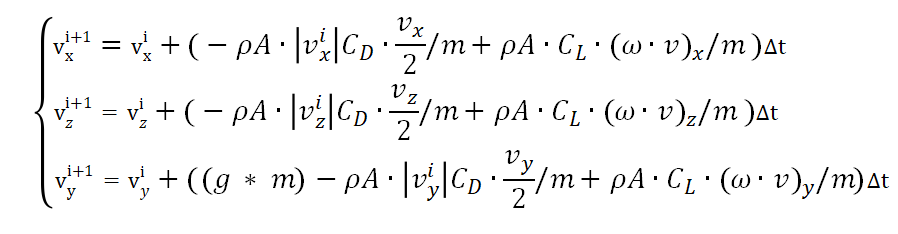
Under rörelsen av ett sfäriskt föremål i verkliga förhållanden möter det ett visst motstånd i sin väg - detta motstånd beror på summan av två komponenter: objektets tangentiella friktion i mediet, vars vektorer riktas längs kroppens yta och tryckkrafterna riktade längs det normala mot ytan. Den motstånd kallas för Luftmotstånd och kan beräknas på detta sätt:

****

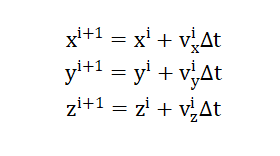
Under denna rörelse, när gasen (luften) strömmar runt objektet, bildas en kraft som verkar på kroppen och riktas vinkelrätt mot flödesriktningen. Ett rörligt objekt skapar en virvel rörelse runt sig själv-detta kallas kollektivt Magnus-effekten som kan beräknas på detta sätt:



Från dessa krafter är ett system av ekvationer konstruerat som beskriver objektets rörelse. Detta kan göras med hjälp av Eulers stegmetoden (Naturvetenskap, 2019-11-15), där dess hastighet beräknas för varje enskild axel. Med hjälp av denna hastighet kan vi få objektets koordinater i n-tidsintervallet. Detta är beräkningen av hastigheten för varje axel med hjälp av Euler-metoden :



Här är beräkningen av objekt positionen med dess acceleration :



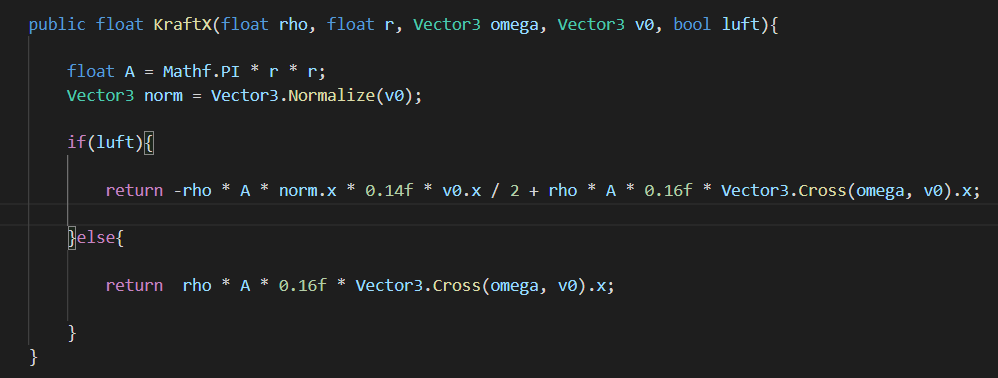
## 2.1 Material

## Spelmotor Unity som är en miljö för utveckling av dataspel. Unity är användarvänligheten som gör skapandet av spel så enkelt och bekvämt som möjligt, och multiplatform motorn gör det möjligt för spelutvecklare att täcka så många spel plattformar och operativsystem som möjligt. En av Unitys styrkor är tyger, kroppens fysik och komplexa animationer. Motorn stöder flera populära programmeringsspråk, men är ursprungligen skrivet i C#, liksom de flesta av extra verktygen som kan förenklar arbetet.

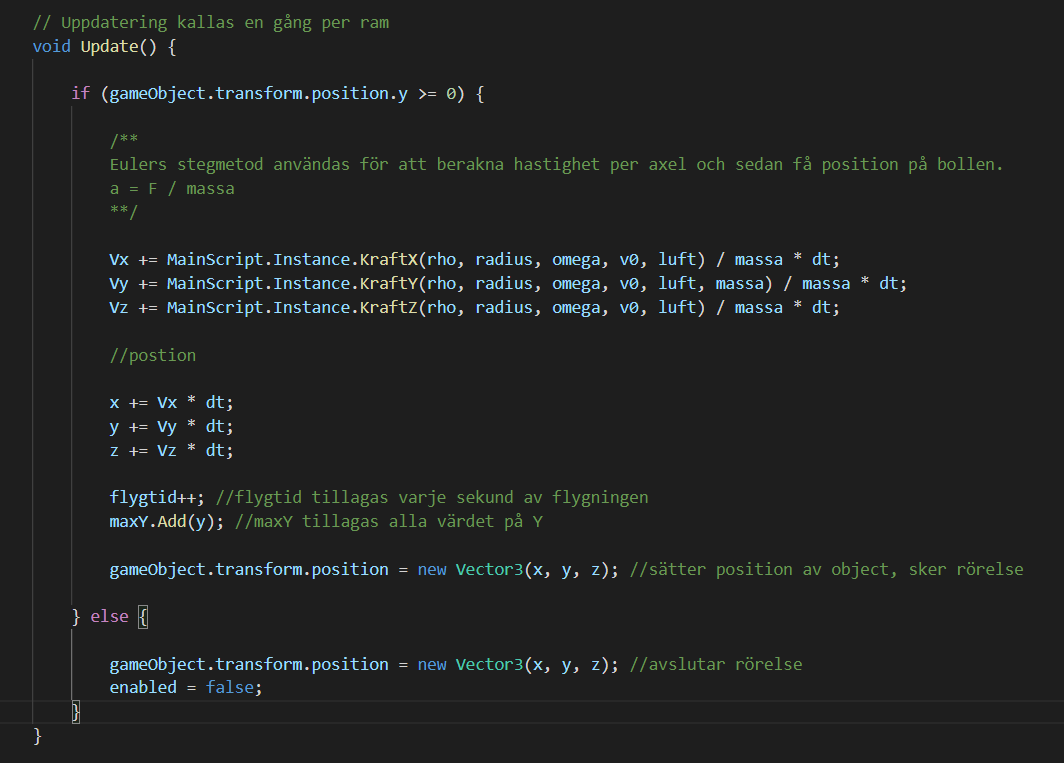
## 2.2 Metod

Implementeringen av koden kan se annorlunda ut beroende på kodens författare. Koden som skrivs i denna uppsats är helt skriven av mig. Innan det önskade resultatet uppnåddes ändrades koden upprepade gånger och testades.

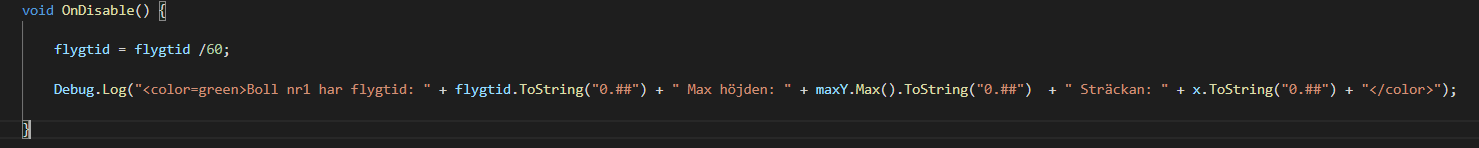
För bättre strukturering av koden och bekvämligheten har jag delat koden i 2 komponenter. Den första delen är databehandling och beräkning av objektets rörelse, och den andra delen är användningen av de bearbetade uppgifterna för att simulera objektets flygning. Alla data lagras i varje objekt separat för att ändra olika parametrar för enskilda objekt. Den del av koden som är ansvarig för beräkning och behandling av data ser ut så här :



Som exempel jag tar upp beräkningen på x-axeln, där finns en metod som heter KraftX(). Den tar data som lufttätheten, objekt radien, omega-vektorn (vinkelhastighet vektor) och den ursprungliga hastighetsvektorn. Den sista parametern är en variabel som beräknar kraften med eller utan luftmotstånd. För att tillämpa denna metod måste vi använda en funktion inbyggd i spelmotorn. Denna funktion kallas Update() se (Unity, 2021-03-10) och sker varje ram.



I bilden ovan syns metoden Update() och beräkningen av hastigheten och objektets position med hjälp av Euler-metoden. Flygning sker endast när objektets y-axelposition är större än eller lika med noll. Variablerna flygtid och maxY kommer senare att användas som ett resultat av experiment. Flygtiden läggs till varje sekund och den maximala höjden registreras i array. I slutet rör sig objektet och om höjden på y-axeln är mindre än noll, stannar rörelsen. När objektet stannar, matas alla insamlade data ut till loggen se (Unity, 2021-03-10).



Den fullständiga och uppdaterade koden finns i bilagor.

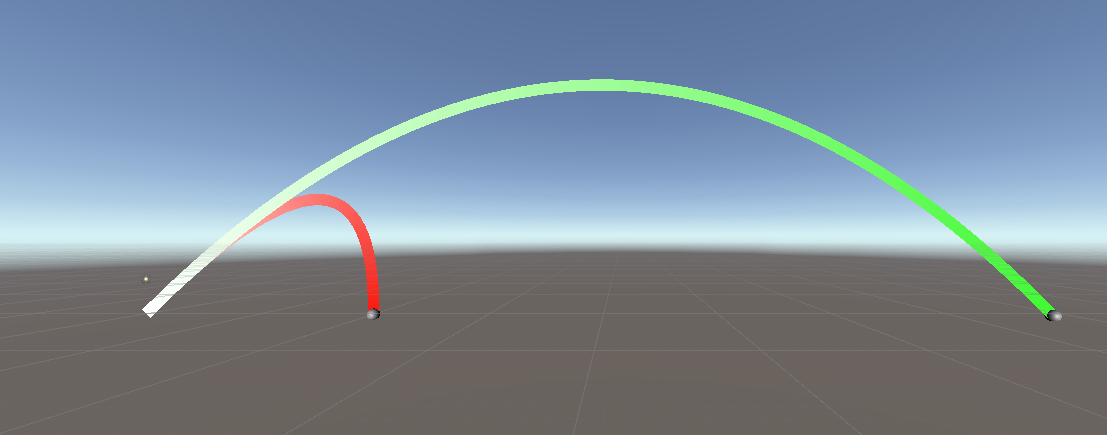
# 

# 3 Resultat och analys

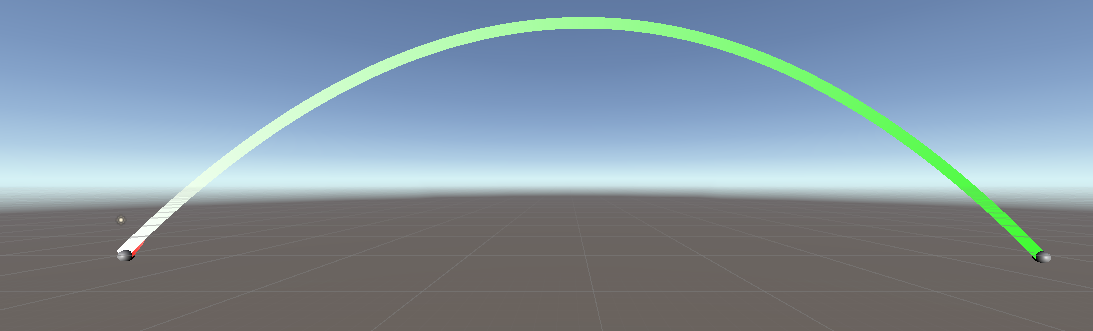
**3.1 Luftmotstånd simulering**

|  |  |  |  |
| --- | --- | --- | --- |
|  | Klassisk simulering  **massa = 0.3056 (kg)**  **radius = 0.105 (m)**  **luft densitet = 1.2 (kg/m3)** | Stor radius simulering  **massa = 0.3056 (kg)**  **radius = 0.905 (m)**  **luft densitet = 1.2 (kg/m3)** | Vatten densitet simulering  **massa = 0.3056 (kg)**  **radius = 0.105(m)**  **luft densitet = 997 (kg/m3)** |
| Flyg tid **(t, sec)** | 2,39 | 1,39 | 0,55 |
| Max höjden **(h, m)** | 20,01 | 9,99 | 1,55 |
| Sträcka **(s, m)** | 79,31 | 19,89 | 0,49 |
| Medelhastighet **(Vm, m/s)** | 22,71 | 15,30 | 13,35 |

Grön boll är alltid klassisk simulering. Starthastighet Vx = 20 & Vy = 20 & Vz = 0 (m/s).



Stor radius simulering.

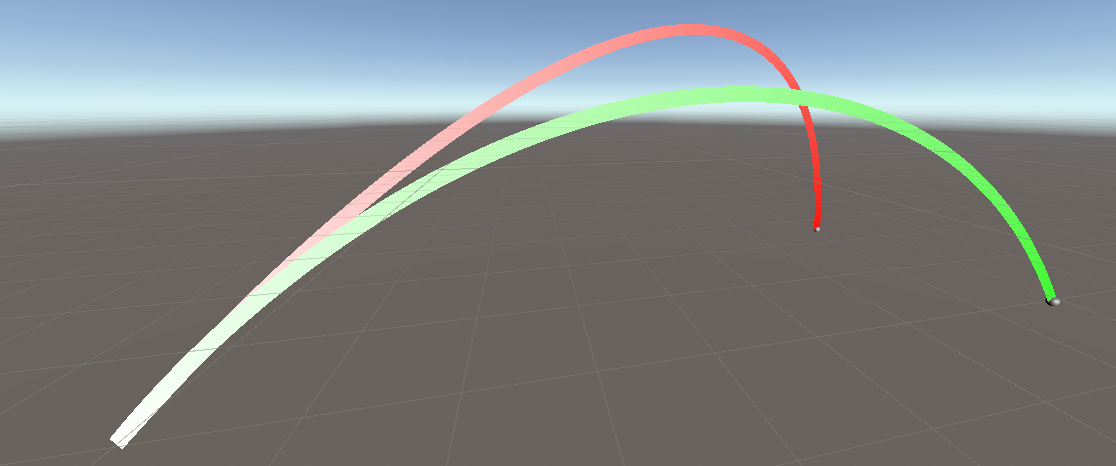


Vatten densitet simulering.

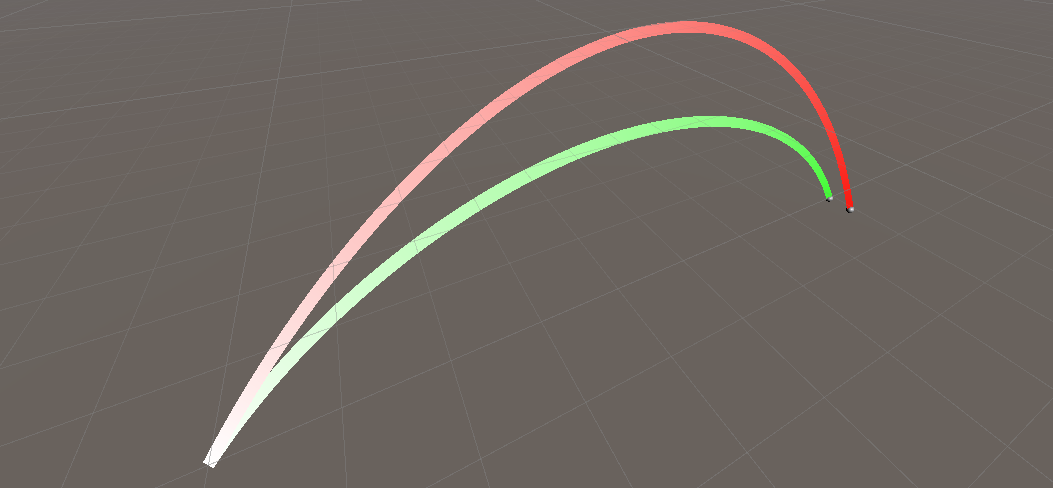
# 3.2 Magnus kraft simulering

|  |  |  |  |
| --- | --- | --- | --- |
| Utgångshastighet & vinkel/omega hastighet | Flyg tid (t, sec) | Max höjden (h, m) | Sträcka (s, m) |
| Vx= 20; Vy = 20; Vz = 0; | 2.47 | 20.01 | 79.31 |
| Wx = 0; Wy = 0; Wz = 0; |
| Vx= 20; Vy = 20; Vz = 5; | 3.07 | 25.65 | 72.32 |
| Wx = 0; Wy = 0; Wz = 5; |
| Vx= 20; Vy = 20; Vz = 5; | 2.98 | 25.65 | 79.55 |
| Wx = 0; Wy = 5; Wz = 5; |

Alla simuleringar hade samma massa, radius och luft densitet. Grön boll är alltid 2-dimensionell rörelse.



Bollen åt vänster simulering.



Bollen åt höger simulering.

# 

# 

# 

# 4 Diskussion och slutsats

**4.1 Fysiska krafter och dess verkan**

**4.1.1 Luftmotstånd**

När ett föremål rör sig på ett ytan eller i luften det skapas krafter som förhindrar det. Det är motstående krafter som skapar drag. Drag påverkar objektets rörelse och saktar ner hastigheten. Detta kan ses i de tre tester som utförts. I det första testet objekt har en ökad radie, vilket leder till en ökning av området som påverkas av luftmotstånd. Ju större området desto större kraft som påverkar den och desto lägre är hastighet. Detta bevisas i "Stor radius simulering". Den genomsnittliga hastigheten för klassisk flygning är 22 m / s när den genomsnittliga hastigheten för flygning med en ökad radie når endast 15.

Förutom området kan hastigheten också påverkas av densiteten hos den miljö där objektet befinner sig. Detta visas i det andra testet där objektet flög i en miljö med densitet som är karakteristisk för vatten. Avståndet, maximal höjd, flygtid och medelhastighet är betydligt mindre än för ett objekt som rör sig i ett utrymme med en medium densitet karakteristisk för luft.

**4.1.2 Magnus Kraft**

Det är en lyftkraft som verkar på en roterande kropp när ett flöde av vätska eller gas strömmar runt den. Denna effekt uppstår som ett resultat av tryckskillnaden på objektets väggar på grund av olika lufthastigheter. Den resulterande obalansen gör att objektet avviker, vilket i sin tur demonstreras i testerna.

**4.2 Egna reflektioner**

**4.2.1 För- och nackdelar med metod**

I det arbetet utfördes inga riktiga tester, alla resultat beskrivs teoretiskt. Det finns inte möjlighet för att genomföra verkliga tester. Alla resultat erhålls med hjälp av programvara och teoretisk kunskap av mänskligheten i denna riktning. Verkliga, upprepade tester gör att du kan bättre och mer exakt studera processen med objekt rörelse och få mer exakta resultat.

Det fanns många problem i detta arbete. Relaterat till både teoretisk och mjukvaru delen. Först var problemen relaterade till byggandet av en matematisk modell. Sedan fanns det problem med genomförandet och insamlingen av all nödvändig information från testen.

**4.2.2 Egna tankar**

Teorin som används i detta dokument går utöver fysik 2 och 3. Vilket är en viss svårighet att förstå och bygga en matematisk modell. Eftersom det fanns svårigheter med byggandet av en matematisk modell fanns det också svårigheter med genomförandet av programdelen.

I de första testerna flög objektet upp utan att stoppa. I flera andra tester flög objektet ner i marken och korrigerade sedan sin rörelse. Efter ett stort antal problem och problemlösningar uppträdde en fungerande rörelsesimulering av ett sfäriskt objekt.

**4.3 Slutsatser**

Under arbetets gång byggdes en matematisk modell av en sfäriskt objekt flygningen rörelse och implementeras med hjälp av spelmotorn (Unity). Jag skulle också vilja notera att arbetet kan innehålla icke-kritiska fel, som i allmänhet inte uttömmer experimentets väsen. Möjliga fel som tillåts i arbetet kan vara i programkoden och matematiska modellen.

Den viktigaste delen i rörelsen av ett objekt spelas av närvaron av en gas eller vätska i mediet. Det är detta faktum som genererar motståndets krafter och Magnus-effektens utseende. Där kraften av luftmotstånd saktar ner objektet och därigenom förändrar objektets beteende och magnus-effekten påverkar flygbanan. Tillsammans kan varje faktor påverka objektets bana och beteende på sin egen väg. Avslutningsvis alla dessa faktorer tillsammans är rörelsesimulering av ett sfäriskt objekt.

# 4. Källförteckning

**4.1 Otryckta källor**

(1) Fysikguiden, Newtons andra lag, <https://fysikguiden.se/newtons-andra-lag/>, 2021-03-14

(2) Naturvetenskap, Eulers stegmetod, <https://www.naturvetenskap.org/matematik/differentialekvationer/eulers-stegmetod/>, 2019-11-15

(3) Unity Dokumentation, MonoBehaviour.Uppdate(), <https://docs.unity3d.com/ScriptReference/MonoBehaviour.Update.html>, 2021-03-10

(4) Unity Dokumentation, Debug.Log(), <https://docs.unity3d.com/ScriptReference/Debug.Log.html>, 2021-03-10

(5) Wikipedia, Magnuseffekten, <https://sv.wikipedia.org/wiki/Magnuseffekten>, 2021-01-28

**4.2 Tryckta källor**

(1) Alphonce, R. Bergström, L. Gunnvald, P. Johansson, E. Nilsson, R. (2013). *Heureka! Fysik 3.* Stockholm: Natur & Kultur.

# 5. Bilagor

Nedan finns länk med kod. Där i MainScript.cs finns viktigaste beräkningar av rörelse. I ball\_1.cs och ball\_2.cs finns variabler som styr rörelse och kod som samlar data för resultatanalys.

https://github.com/DeLanau/GYarbete