

eman ta zabal zazu



Universidad  
del País Vasco

Euskal Herriko  
Unibertsitatea



ZTF-FCT

Zientzia eta Teknologia Fakultatea  
Facultad de Ciencia y Tecnología



Gradu Amaierako Lana / Trabajo Fin de Grado  
Fisikako Gradua / Grado en Física

## Simulazioak Fisikaren Irakaskuntzan

Egilea/Autor:  
Peio Iñurriagarro Rodriguez  
Zuzendaria/Director/a:  
Juan Mari Aguirregabiria



# Gaien Aurkibidea

Aurkibide orokorra . . . . .	3
Irudien Zerrenda . . . . .	5
<b>1 Sarrera</b>	<b>6</b>
1.1 Zer da Easy Java Simulations (EJS) . . . . .	6
<b>2 Easy Java Simulations-ren erabilera</b>	<b>8</b>
2.1 Deskribapena . . . . .	9
2.2 Modeloa . . . . .	9
2.3 Bista . . . . .	11
2.4 Simulazioaren exekuzioa . . . . .	11
<b>3 Batxilergo mailako simulazioak</b>	<b>12</b>
3.1 Tiro parabolikoa . . . . .	12
3.1.1 Problemaren aurkezpena . . . . .	12
3.1.2 Azalpen teorikoa . . . . .	12
3.1.3 Simulazioa . . . . .	15
3.1.4 Ariketak: . . . . .	18
3.2 Pendulu sinplea . . . . .	19
3.2.1 Problemaren aurkezpena . . . . .	19
3.2.2 Azalpen teorikoa . . . . .	19
3.2.3 Simulazioa . . . . .	20
3.2.4 Ariketak . . . . .	23
<b>4 Unibertsitate mailako simulazioak</b>	<b>24</b>
4.1 Eguzki Sistema . . . . .	24
4.1.1 Problemaren aurkezpena . . . . .	24
4.1.2 Azalpen teorikoa . . . . .	24
4.1.3 Simulazioa . . . . .	26
4.1.4 Ariketak . . . . .	30
4.2 Atwood-en makina kulunkatzailea ( <i>Swinging Atwood's machine</i> )	31
4.2.1 Problemaren aurkezpena . . . . .	31

---

4.2.2	Azalpen teorikoa . . . . .	31
4.2.3	Simulazioa . . . . .	34
4.2.4	Ariketak . . . . .	37
<b>5</b>	<b>Ondorioak</b>	<b>39</b>
	<b>Bibliografia</b>	<b>41</b>

# Irudien Zerrenda

2.1	Programa irekitzean agertzen diren leihoak. . . . .	8
3.1	Tiro parabolikoaren problemaren eskema. . . . .	13
3.2	Tiro parabolikoaren problemaren eskema marruskadurarekin. . . . .	14
3.3	Tiro parabolikoaren simulazioa martxan. . . . .	16
3.4	Pendulu sinplearen eskema. . . . .	19
3.5	Pendulu sinplearen simulazioaren exekuzioa. . . . .	22
4.1	Bi gorputzen problemaren eskema. . . . .	25
4.2	Eguzki Sistemaren simulazioaren pantaila. . . . .	27
4.3	Eguzki Sistemaren simulazioa. . . . .	30
4.4	Atwood-en makina kulunkatzailearen problemaren eskema. . . . .	31
4.5	Atwood-en makina kulunkatzailearen simulazioaren pantaila. . . . .	35

# Kapitulua 1

## Sarrera

Fisika irakasterako orduan ordenagailuaren bidezko simulazioak erabiltzea baliabide egoki bat izan daiteke, klaseak erakargarriagoak egiten dituztelako eta aztertzen ari den fenomenoari buruzko ezagutzan sakondu daitekeelako era bisual baten.

Lan honen helburua da fisikaren irakaskuntzarako simulazio didaktiko batzuk egitea. Horretarako Easy Java Simulations programa erabili egin dut hainbat fenomeno fisiko simulatzeko, pare bat batxilergo mailarako (tiro parabolikoa eta pendulu sinplea) eta beste bi unibertsitate mailarako (N gorputzen problema eta Atwood-en makina kulunkatzailea). Egindako simulazioak deskargatzeko simulazio bakoitzaren “*Simulazioa*” azpiatalaren hasieran deskargatzeko link-a dago. Bertan klik egin eta irekitzen den web orrian *Descargar a tu ordenador* klikatu.

Simulazioak ikusi ahal izateko JAVA instalatuta egon behar da, JAVA deskargatzeko tokia: <https://www.java.com/es/download/>.

Easy Java Simulations programaz gain, txostenean agertzen diren fenomeno-en eskemen irudiak egiteko <https://www.draw.io/> web orria eta GIMP programa erabili egin ditut.

### 1.1 Zer da Easy Java Simulations (EJS)

Easy Java Simulations (EJS) ordenagailu bidez simulazioak sortzeko diseinatuta dagoen doako software tresna bat da. Programa honen helburua da erabiltzaileak euren simulazioak sortu ahal izatea era erraz baten.

Ordenagailu bidezko simulazio bat ordenagailu programa bat da, fenomeno natural bat erakusten saiatzen dena bere egoera posibleak erakutsiz. Egoera posible hauek denborarekin aldatzen diren aldagai batzuen bidez deskribatzen dira.

EJS programaren abantaila da berak automatikoki sortzen duela beharrezko kodea, ahalbidetuz erabiltzailea gehiago zentratzea simulatu nahi den fenomenoaren deskribapenean beharrezko programazio xehetasun teknikoetan baino. Beraz, EJS-k simulazioa sortzerakoan erabiltzaileak egin beharreko lana sinplifikatzen du. Alde batetik *bistaren* sorkuntza (simulazioaren atal grafikoa) sinplifikatu egiten duelako, normalean programazio ezagutza tekniko handia eskatzen duena. Eta beste alde batetik, simulazioaren *modelo* (fenomenoaren deskribapen zientifikoa) sortzeko egitura sinplifikatu bat eskaintzen duelako.

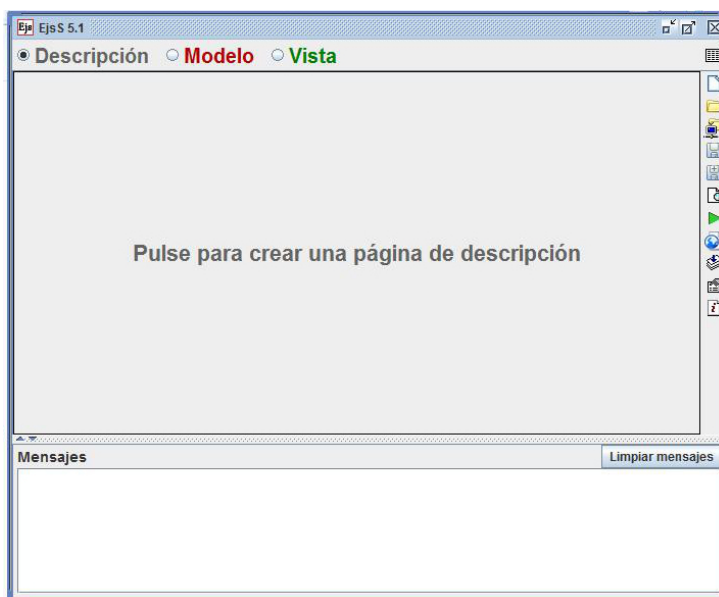
Abantailetak beste bat da JAVA erabiltzen duenez sortutako “jar” fitxategiak erabil daitezkeela edozein sistema eragileetan. Baita ere posiblea da simulazioak gordetzea interneteko web orri baten ikusi ahal izateko moduan. Gainera, simulazioaren kodea programa exekutagarriarekin batera jar daiteke, horrela erabiltzaileak bere gustura moldatzeko aukera izanik.

Azkenik, EJS-ren aldeko beste ezaugarri bi dira programak lehen ordenako ekuazio diferentzialak ebazteko badituela alde zuzenetik prestatuta hainbat metodo (ikusi 2.2 atalaren *Evolución* zatia) eta EJS oso erabilgarria dela soluzio analitikorik gabeko sistema ez-linealak aztertzerako orduan.

## Kapitulua 2

# Easy Java Simulations-ren erabilera

Programa bere web gunetik deskarga daiteke <http://www.um.es/fem/EjsWiki/Main/Download>. Programa irekitzean, interfasean lau gune daudela ikus daiteke (ikus 2.1 irudia):



Irudia 2.1: Programa irekitzean agertzen diren leihoak.

- Goiko gunea: Lan panelaren aukeraketa. *Descripción*, *Modelo* eta *Vista* atalak aukeratzeko tokia.
- Erdiko gunea: Lan panela. Lan egingo dugun tokia.



- Eskumako gunea: Zeregin barra. Simulazio berria sortzeko, simulazio bat kargatzeko, gordetzeko, exekutatzeko... aukerak daude hemen.
- Beheko gunea: Mezu gunea. Simulazioa sortzerakoan emandako agin-  
duen emaitzaren informazioa agertuko da hemen.

Simulazioak hiru atal nagusi dauzka: Deskribapena, modeloa eta bista. *Descripción* atalean egileak simulazioari buruzko idatzitako azalpenak agertzen dira. *Modelo* atalean simulatu nahi den fenomenoaren eredua sortu egiten da. Azkenik, *Vista* atalean simulazioaren interfaze grafikoa diseinatu egiten da. Atal hauetako bakoitzean lan egiteko “Lan panelaren aukeraketan” dauden hiru botoiak erabili behar dira. Botoi hauetako bat aukeratzean, atal horri dagokion “Lan panela” agertzen da, bertan atal horri dagozkion aldaketak egin ahal izateko. Orain hiru atal hauek pixka bat sakonago aztertuko dira:

## 2.1 Deskribapena

Lehen esan bezala atal honetan simulazioari buruzko azalpenak eman ditzake egileak, bai fenomenoari buruzko azalpenak bai simulazioa nola erabiltzeari buruzkoa adibidez. Orri hauek sor daitezke HTML kodea idatziz edo WYSIWYG editore batekin.

## 2.2 Modeloa

Atal honetan simulatu nahi den fenomenoaren eredua sortu egiten da. Fenomeno baten eredu bat sortzerako orduan fenomenoaren aldagaiak zeintzuk diren definitu behar dira, hauei hasierako une bateko balioak eman eta eboluzioa zehazten duten legeak zehaztu behar dira. Aldagaiak izan daitezke egoera aldagaiak (fenomenoaren egoera zehazten dutenak) edo beste edonolako parametroak, denboran zehar alda daitezkeenak edo ez.

Ereduaren egoera une bateko aldagaien balioek zehaztuko dute. Egoera alda daiteke bai simulazioaren barne dinamikagatik (*eboluzioa* deritzona) edo simulazioaren erabiltzaileak eskuz aldagaien balioa aldatzean. Baita egon daitezke *loturak* aldagaien artean, hau da aldagai batzuk esplizituki izatea beste batzuen menpekoak eta, ondorioz, azken hauen aldaketak ekartzea besteak ere aldatzea.

Beraz, simulazioaren eredua eraikitzeko zehaztu behar da zeintzuk diren beharrezkoak diren aldagaiak, hasierako egoera, eboluzio ekuazioak eta lotura ekuazioak. Lan panelaren aukeraketan *Modelo* aukeratzean, honi dagokion lan eremua agertzen da, hainbat orri dituen fenomeno bat erduztatzeko behar diren elementuak zehaztu ahal izateko: *Variables*, *Inicialización*, *Evolución*, *Relaciones fijas* eta *Propio*.

*Variables* orrian, simulazioko aldagaiak eta parametroak definitu behar dira. Hauen izaera ere zehaztu behar da: erreala (**double**), osoa (**int**), boolearra (**boolean**), karakterea (**String**) edo objektua (**Object**). Baita ere dimentsioa zehaztu daiteke eta komentario txiki bat idatzi azalpen batekin. Orri honetan bertan aldagaiei ere eman ahal zaie hasierako balio bat, hau oso konplexua ez bada.

Hastapen baldintza konplexuagoak idazteko, *Inicialización* orria dago eskuragarri. Bertan nahi diren hastapen baldintzak sortzen dituen Java kodea idatz daiteke.

*Evolución* orrian simulazioaren eboluzioa zehaztu behar da eboluzio ekuazioak ebatziz. Eboluzio ekuazioak ebazteko bi aukera ditugu: Simulazioaren sortzaileak Java kode orri bat sortzea ekuazioak ebatziz edo “EDO” editorea (ekuazio diferentzial arruntetarako editorea) erabiltzea.

Ekuazio diferentzialak ebazteko beste aukera da “EDO” editorea erabiltzea.

Lehenengo aukeran simulazioaren sortzaileak idatzi behar du eboluzio ekuazioak ebazteko beharrezkoa den Java kodea. Bigarren aukeran EJS-ak automatikoki sortuko du beharrezko Java kodea, programak ekuazio diferentzialak ebazteko eskaintzen dituen metodoen artean erabiltzaileak “EDO” editorearen bidez aukeratutakoa inplementatuz. Baita ere posiblea da bi editoreak aldi berean erabiltzea orri bat baino gehiago sortuz *Evolución* orri honetan. Deribatu egiten diren aldagaiak dimentsiodunak edo unidimentsionalak izan daitezke, baina beti **double** motakoak.

“EDO” editorea erabiltzeko, lehenengo eta behin aldagai independentea zein den esan behar zaio EJS-ri eta integrazio pausua zehaztu behar da, konstante baten edo double motako aldagai baten bidez. Azken honekin ekuazioak ebatzi ostean eboluzio pausu bakoitzean aldagai independenteak duen aldaketa zehazten da. Arrazoi honengatik aldagai independente berdina erabiltzen duten ekuazio guztiak orri berean idatzi behar dira. Jarraian simulatu nahi den sistemaren ekuazio diferentzialak sartu behar dira. Ekuazioak

lehen ordenako ekuazio diferentzialak izan behar dira. Gero, ebazpenerako erabiliko den algoritmo numerikoa aukera daiteke eskaintzen den zerrendatik bat aukeratuz eta programak sortuko du beharrezkoa den kodea.

*Relaciones fijas* panelean lotura ekuazioak idatzi behar dira dagokion Java kodea idatziz.

*Propio* panelean erabiltzaileak nahi dituen Java metodoak idatz daitezke (beste programazio lengoaietako funtzio edo subrutinak) Java kode baten bidez gero simulazioko beste toki batzuetan erabiltzeko. Baita ere existitzen dira aurretik zehaztutako metodoak, EJS-ren berezkoak diren batzuk edo kanpoko librerietako metodoak, adibidez Javaren libreria matematikoenak.

## 2.3 Bista

Atal honetan simulazioaren interfaze grafikoa diseinatu egiten da zuhaitz estruktura baten bidez, bertan elementu grafikoak kokatuz. EJS-ek hainbat elementu mota desberdin ematen ditu aukeran, hala nola 2D edo 3D inguruneak, malgukiak, bolak, bektoreak...

## 2.4 Simulazioaren exekuzioa

Simulazioa exekutatzean hau da gertatzen dena: Lehenengo aldagaiak sortu egiten dira eta balioak zehazten dira hastapen baldintzen bidez. Ondoren lotura ekuazioak ebaluatu egiten dira. Gero, bista sortu egiten da simulazioa hasierako egoeran utziz. Hemen bi aukera daude, alde batetik eboluzio ekuazioak ebaluatu egiten dira eta hauen jarraian lotura ekuazioak modeloa hurrengo denbora pausura eramanez eta bista berrituz; edo bestalde erabiltzaileak aldagai baten balio alda dezake, honen jarraian lotura ekuazioak ebaluatzen dira berriro eta bista berritzen da egoera berri bat lortuz denbora une berean.

# Kapitulua 3

## Batxilergo mailako simulazioak

Atal honetan batxilergo mailarako burututako simulazioak aztertuko dira: Tiro parabolikoa eta pendulu sinplea.

### 3.1 Tiro parabolikoa

#### 3.1.1 Problemaren aurkezpena

Eremu grabitatorio uniforme baten eta marruskadurarik gabeko egoeran objektu baten jaurtiketari tiro parabolikoa deritzo. Demagun hasieran  $m$  masako objektua pausagunean dagoela eta  $\vec{v}_0$  abiadura bat eman zaiola horizontalarekiko  $\alpha$  angelu batekin, helburuak izango dira jakitea noraino helduko den objektua, hartzen duen altuera maximoa eta abar. Hasieran ez da marruskadurarik kontuan hartuko baina gero simulazioa hedatuko da marruskaduradun kasura.

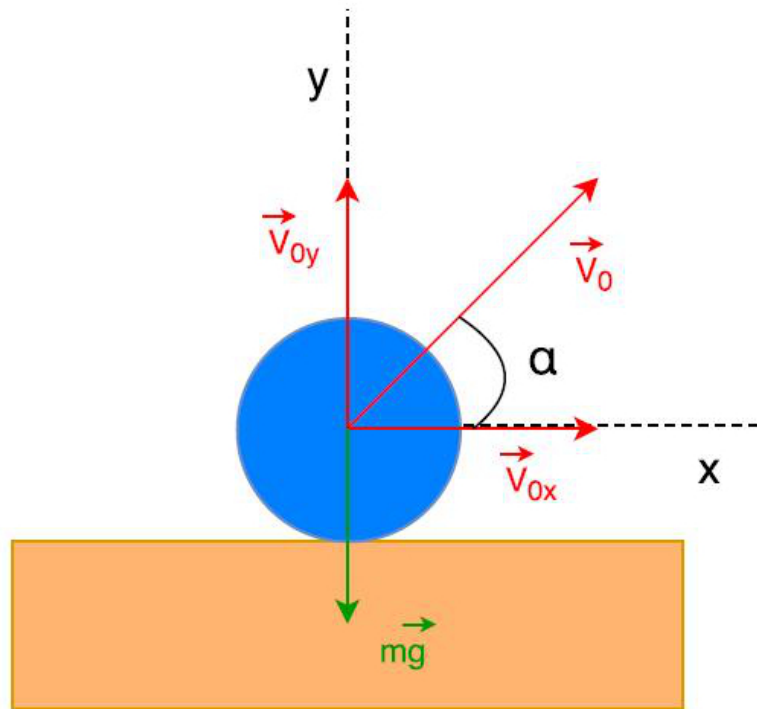
#### 3.1.2 Azalpen teorikoa

Hurrengo irudian (3.1 irudia) marruskadurarik gabeko kasuko eskema bat ikus daiteke.

Higidura ekuazioak Newton-en bigarren legeak emango dizkigu,  $\vec{F} = m\vec{a}$ , horretarako ardatz bakoitzeko indarren batura egin behar da. Beraz, hauek dira kasu honetan higidura ekuazioak:

$$x \text{ ardatzean : } m\ddot{x} = 0 \quad (3.1)$$

$$y \text{ ardatzean : } m\ddot{y} = -mg \quad (3.2)$$



Irudia 3.1: Tiro parabolikoaren problemaren eskema.

Non bi puntuek denborarekiko bigarren deribatua adierazten duten. Programak lehen ordenako ekuazioak behar dituzenez, honako baliabidea erabiliko da ordena murrizteko 3.1 eta 3.2 higidura ekuazioetan: Definitzen badira  $v_x = \dot{x}$  eta  $v_y = \dot{y}$  hauek dira ebatzi beharko diren lehen ordenako eboluzio ekuazioak:

$$\dot{v}_x = 0 \quad (3.3)$$

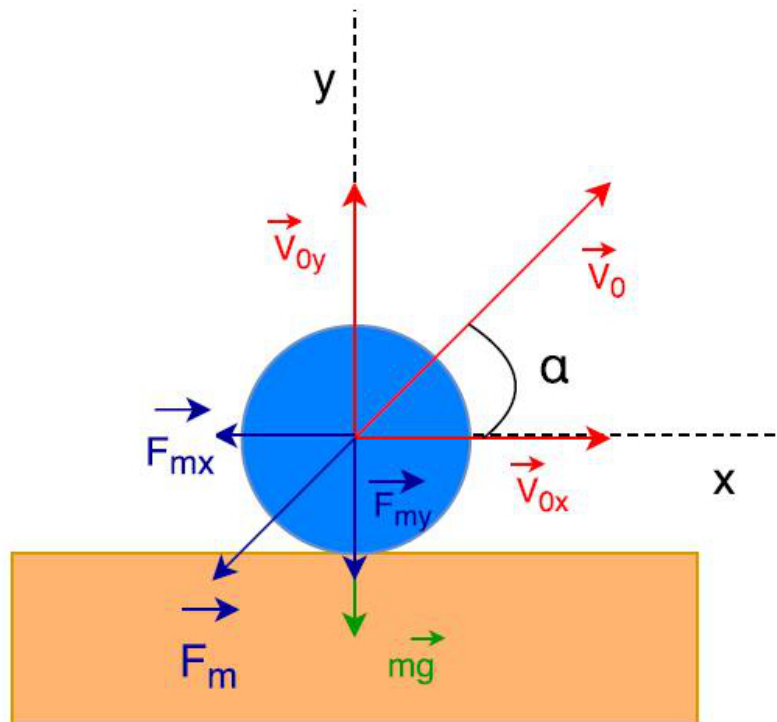
$$\dot{x} = v_x \quad (3.4)$$

$$\dot{v}_y = -g \quad (3.5)$$

$$\dot{y} = v_y \quad (3.6)$$

Orain suposatzen badugu objektua fluido baten barnean dagoela murgilduta eta *marruskadura* dugula ( $\vec{F}_m$  da marruskadura indarra), higidura ekuazioak aldatuko dira. Kasu honetan objektuaren gaineko indarren eskema 3.2 irudian ikus daiteke.

Orduan, egoera horretan hauek dira higidura ekuazioak:



Irudia 3.2: Tiro parabolikoaren problemaren eskema marruskadurarekin.

$$x \text{ ardatzean : } m\ddot{x} = F_{mx} \quad (3.7)$$

$$y \text{ ardatzean : } m\ddot{y} = -mg + F_{my} \quad (3.8)$$

3.7 eta 3.8 ekuazioetan ez dira jarri zeinu negatiboak marruskadura indarraren osagaietan marruskadura indarraren definizioaren barruan sartu direlako. Ezaguna denez, marruskadura hau da objektua fluido baten murgilduta egoteagatik biskositatearen ondorioz higiduraren kontra agertzen den indar bat, abiaduraren menpekoa dena. Marruskadura indarra kasu honetan honako adierazpenak emango du:  $\vec{F}_m = -mb|\vec{v}|\vec{v}$ , non  $b$  arrastrearen koefizientea den. Kasu honetan ere aurreko kasuko definizioak erabiliz lehen ordenako honako ekuazio diferentzialak lortzen dira:

$$\dot{v}_x = F_{mx} \quad (3.9)$$

$$\dot{x} = v_x \quad (3.10)$$

$$\dot{v}_y = -g + F_{my} \quad (3.11)$$

$$\dot{y} = v_y \quad (3.12)$$

Marruskadura indar honen eraginaren ondorioz sistema jada ez da tribiala izango eta hau aztertzeke EJS oso erabilgarria izango da, fenomenoak era “simple” baten aztertzea ahalbidetuz.

### 3.1.3 Simulazioa

#### Simulazioa deskargatzeko:

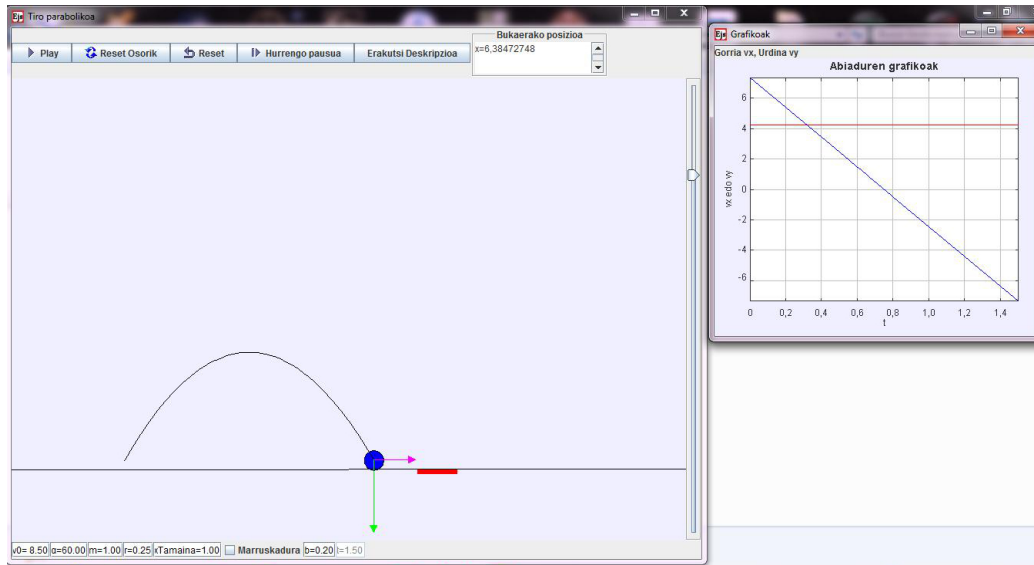
[https://mega.co.nz/#!WBclQaaY!ME4wRyspDOZsYBdvUOLKNBiJW6QBuP\\_igCHF5\\_z9RZO](https://mega.co.nz/#!WBclQaaY!ME4wRyspDOZsYBdvUOLKNBiJW6QBuP_igCHF5_z9RZO)

Simulazioan jaurtitzen den objektua zirkulu bat izango da eta marruskadurarik ez badago programak hasierako egoera batetik hasita 3.3, 3.4, 3.5 eta 3.6 ekuazioak ebatziko ditu; marruskadura badago ordea 3.9, 3.10, 3.11 eta 3.12 ekuazioak ebatziko ditu. EJS programak lehen ordenako ekuazio diferentzialak ebazteko ematen dituen aukeretatik Dormand-Prince8(53) metodoa erabili egin da. Simulazioa gelditu egingo da objektuak lurra berriz jotzen duenean.

Defektuzko hasierako egoera da (honekin esan nahi dut hau dela programa exekutatzeko agertzen den hasierako egoera) zirkulua  $(x, y) = (0, r)$  puntuan egotea, non  $r$  zirkuluaren erradioa (hasieran 0.25), hasierako abiadura  $v_0 = 8.5$  eta hasierako angelua  $\alpha = 60^\circ$  diren. Baita ere ikus daiteke marka gorri bat dagoela lurrean, defektuz markak 1.0-ko tamaina du eta honen zentroa  $x = 8.0$  puntuan dago kokatuta. Defektuz marruskadura ez da kontuan hartuko. Simulazioaren pantaila hurrengo irudian ikus daiteke (3.3 irudia). Irudi honetan simulazioaren exekuzio bat ikusten da, defektuzko hasierako balioekin egin eta marruskadurarik gabekoa.

Simulazioak bi pantaila ditu, 3.3 irudiaren ezkerreko leihoa da pantaila nagusia eta bestean grafiko batzuk aurkeztu egiten dira.

Pantaila nagusiaren goialdean simulazioaren exekuzioa kontrolatzeko botoiak daude, behealdean parametroak aldatzeko aukerak daude, erdian fenomenoak irudikatu egiten da eta eskumako haga bertikala zoom-a da. Irudian



Irudia 3.3: Tiro parabolikoaren simulazioa martxan.

agertzen diren geziak abiaduraren osagaiak dira: magenta abiadura horizontala da eta berdea abiadura bertikala, lau aldiz txikituak agertzen dira irudikatuta balioak. Orain botoien eta aukera desberdinen azalpena emango da.

#### Exekuzio botoiak:

- *Play*: Simulazioa martxan jartzen du. Simulazio martxan badago *Play* botoiaren tokian *Stop* botoia egongo da eta azken honek simulazioaren exekuzioa gelditzen du.
- *Reset Osorik*: Botoi honek simulazioaren baldintza guztiak defektuzko hasierakoetara bueltatzen ditu.
- *Reset*: Botoi honek simulazioa exekutatu den hasierako baldintzetara bueltatzeko balio du. Erabiltzaileak hasierako baldintzak aldatu baditu hauetara bueltatuko da.
- *Hurrengo pausua*: Botoi honen bidez denbora pausu bakar bat egiten da aurrera.
- *Erakutsi Deskripzioa*: Botoi honek simulazioaren deskribapen panela erakusten du.
- *Bukaerako posizioa*: Toki honetan agertuko da objektuak lurra jo dueneko x posizioaren balioa.



Parametroen aldaketa:

Ezkerretik hasita eskumara hau da karratu bakoitzak kontrolatzen duen parametroa:

- $v_0$  : Hasieran objektuari ematen zaion abiadura kontrolatzen du.
- $\alpha$  : Objektuaren jaurtiketaren horizontalarekiko hasierako angelua kontrolatzen du.
- $m$  : Objektuaren masa aldatzeko balio du.
- $r$  : Objektuaren erradioa kontrolatzen du.
- $xTaina$  : Marka gorriaren  $x$  norabideko tamaina kontrolatzen du.
- *Marruskadura*: Marruskadura kontuan hartzeko laukia tikatuta egon behar da, bestela ez da kontuan hartzen.
- $b$  : Arrastrearen koefizientea kontrolatzen du.
- $t$  : Simulazioa abiarazi denetik pasatu den denbora. Balio hau ezin du erabiltzaileak aldatu.

Aldaketak egiteko nahi den balio numerikoa jarri behar da eta *Enter* tekla sakatu. Beste aldaketa batzuk egitea ere posiblea da: objektuaren hasierako posizioa alda daiteke, horretarako objektu urdinaren gainean (jaurtigaia) klikatu saguarekin eta askatu gabe arrastratu nahi den tokiraino. Antzeko moduan marka gorriaren posizioa alda daiteke ere. Jaurtigaia  $x$  eta  $y$  bi norabideetan mugi daiteke, baina marka gorria bakarrik  $x$  norabidean. Simulazioan agertzen den marra horizontalak lurzorua adierazten du ( $y = 0$ ), jaurtigaia marra honen azpitik jartzen saiatzerakoan automatikoki lurzoruan jarriko da jaurtigaia berriro, hau da, jaurtigaia luraren barrutik botatzea debekatuta dago simulazioan.

Azkenik, eskuman dagoen barra bertikalak *zoom*-a aldatu egiten du. Osagai desberdinen gainean sagua uzten bada, horri buruzko informazio pixka bat duen laukizuzen bat agertzen da.

Eskumako leihoan grafiko batzuk agertzen dira. Abiaduren grafikoan agertzen den lerro gorria  $v_x$  abiaduraren osagai horizontala da eta lerro urdina  $v_y$  abiaduraren osagai bertikala.

Simulazioa exekutatzean *HasieraTiro* izeneko fitxategi bat sortuko da simulazioa gordeta dagoen karpetan. Hemen gordeko da sistemaren hasierako egoera (justu simulazioa martxan jarri aurretik duguna), defektuzkoa edo aldatua aldaketak egin badira. Gordeko diren aldagaiak hauek dira:  $x, y, v_0, v_x, v_y, alfa, m, r, xMarka, xTamaina, b$  eta  $t$ . Egoera honetara itzuli ahalko gara *Reset* botoiaren bidez.

### 3.1.4 Ariketak:

#### 1. Marruskadurarik gabe:

- (a) Jarri marka gorria nahi den puntuan eta saiatu zaitez parametroak aldatuz jaurtitzen den masa bertara heltzea, hau da, irismena kalkulatzeko. Saiatu baita ere jaurtigaiaren hasierako konfigurazio desberdinetarako: Altuera batetik botaz, hasierako abiaduren osagaien bat nulua egin,... edo bururatzen zaizun edozein konbinazio.
- (b) Zein da abiadura jakin batentzat angelua, jaurtigaiak egindako distantzia maximoa izateko? Agertzen diren parametro guztiek eragina dute?

#### 2. Marruskadurarekin:

- (a) Errepikatu aurreko ariketako galderak. Aldaketarik ba al dago?
- (b) Aztertu baita ere  $b$  parametroaren aldaketak sortzen duen eragina.

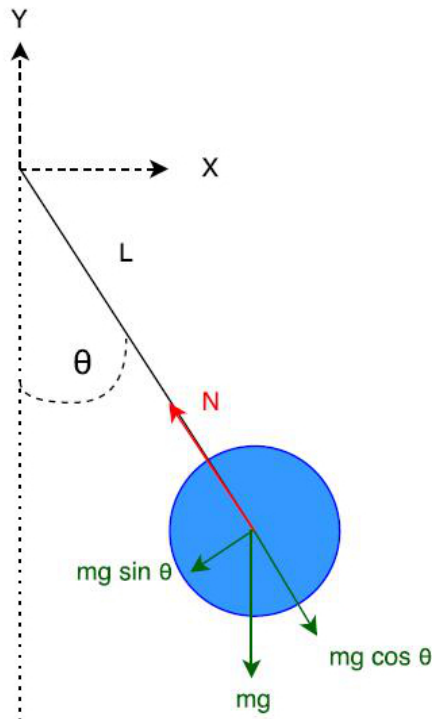
## 3.2 Pendulu simplea

### 3.2.1 Problemaren aurkezpena

Pendulu simple bat osatuta dago mutur batetik eskegita dagoen eta beste muturrean masa bat duen  $l$  luzeradun soka batengatik. Masa angelu bat aldentzen da bertikaletik eta askatu egiten da, posiblea da hasierako abiadurarik gabe askatzea edo hasieran abiadura bat ematea. Simulatutako kasuan objektua  $r$  erradioko zirkulu bat izango da.

### 3.2.2 Azalpen teorikoa

Hurrengo irudian (3.4 irudia) problemaren eskema bat ikus daiteke:



Irudia 3.4: Pendulu simplearen eskema.

Kasu honetan koordinatu polarrak erabili egin dira problema ebazteko. Ez dago higidurarik norabide erradialean, bakarrik dago higidura norabide angeluarrean, beraz, higidura ekuazio bakarra behar da. Norabide horretan dagoen grabitate indarraren osagaia  $mg \sin \theta$  da eta norabide horretan azelerazioa  $l\ddot{\theta}$  da objektua  $l$  erradioko zirkunferentzi baten zehar higitzen

ari delako, non bi puntuek denborarekiko bigarren deribatua adierazten duten. Beraz, Newtonen bigarren legea aplikatuz, hauek dira lortzen ditugun bi ekuazioak:

$$N = mg \cos \theta \quad (3.13)$$

$$mgl\ddot{\theta} = -mg \sin \theta \Rightarrow l\ddot{\theta} = -g \sin \theta \quad (3.14)$$

Beraz, higidura ekuazioa:

$$l\ddot{\theta} = -g \sin \theta \quad (3.15)$$

Zeinu negatiboa dator indar hori beti delako desplazamenduaren kontrako. Ekuazioan agertzen den  $\sin \theta$  faktorea dela eta, ekuazio ez-lineal bat da, eta ondorioz ez da erreza izango beti soluzio analitiko bat lortzea. Normalean, soluzio analitikoak lortzeko angelu txikien hurbilketara jotzen da, hau da,  $\sin \theta \approx \theta$  egitea. Kasu honetan hau da geratzen den ekuazioa:

$$l\ddot{\theta} = -g\theta \quad (3.16)$$

Eta ekuazio hori ezaguna den osziladore harmoniko sinplearen ekuazioa da. Baina simulazioak metodo numerikoak erabiltzen dituen ekuazio ez-lineala (3.15 ekuazioa) baita ere ebazteko gai izango da.

Kasu honetan ere lehen ordenako ekuazio diferentzialak behar direnez,  $\dot{\theta} = \omega$  definizioaren bidez honako lehen ordenako bi ekuazio diferentzialak lortzen dira 3.15 ekuaziotik:

$$l\dot{\omega} = -g \sin \theta \quad (3.17)$$

$$\dot{\theta} = \omega \quad (3.18)$$

### 3.2.3 Simulazioa

#### Simulazioa deskargatzeko:

<https://mega.co.nz/#!OYEF1CAT!wduk5XrLCI6JxKh7JGp179uuTRaUTUI1kWKtH2-jtMU>

Programak hasierako egoeratik hasi eta 3.17 eta 3.18 ekuazioak ebartziko ditu. Aukeratu egin den metodoa Dormand-Prince8(53) metodoa da kasu honetan.

Defektuzko hasierako egoera da pausagunean dagoen 3 masako eta 0.25 erradioko zirkulua, sokaren luzera 2 da eta zirkuluak bertikalarekin osatzen duen hasierako angelua  $25^\circ$ -koa.

Azalpen teorikoaren atalean esan bezala, koordenatu polarrak erabili egin dira higidura ekuazioa lortzeko. Beraz, irudikapena egiteko orduan koordenatu kartesiarretara pasatu behar denez honako transformazioak erabili egin dira:

$$x = l \sin \theta \quad (3.19)$$

$$y = -l \cos \theta \quad (3.20)$$

Posizio bertikaleko minus zeinua dator Y ardatzaren noranzko positiboa goranzkoa delako. Abiadura linealaren osagaiak hauek dira:

$$v_x = l\omega \cos \theta \quad (3.21)$$

$$v_y = l\omega \sin \theta \quad (3.22)$$

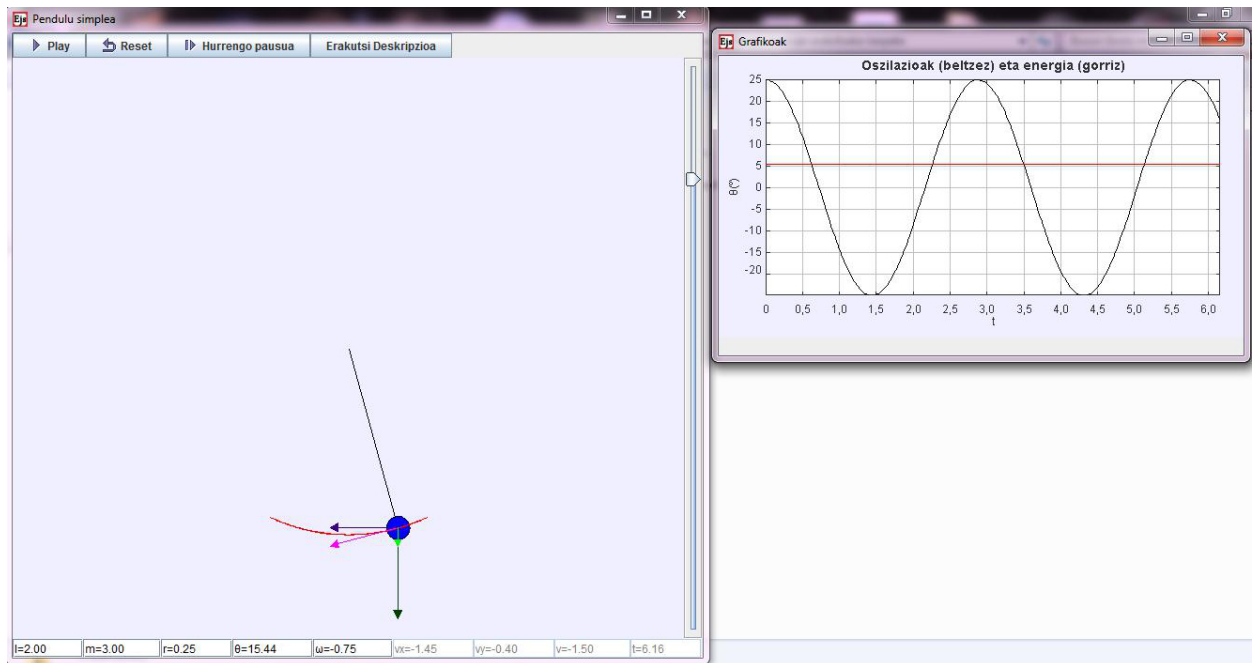
$$v = l\omega \quad (3.23)$$

Hurrengo irudian (3.5 irudia) simulazioaren pantaila ikus daiteke. Simulazioa martxan jarri egin da, hasierako defektuzko egoeratik.

Aurreko tiro parabolikoaren kasuan bezala, simulazioa honek bi pantaila ditu, 3.5 irudiaren ezkerreko leihoa da pantaila nagusia eta bestean grafiko batzuk aurkeztu egiten dira. Kasu honetan ere pantaila nagusiaren goialdean simulazioaren exekuzioa kontrolatzeko botoiak daude behealdean parametroak aldatzeko aukerak dauden bitartean, erdian fenomenoak marraztu egiten da eta eskumako haga bertikala zoom-a da. Orain botoien eta aukera desberdinen azalpena emango da.

#### Exekuzio botoiak:

- *Play*: Simulazioa martxan jartzen du. Simulazio martxan badago *Play* botoiaren tokian *Stop* botoia egongo da eta azken honek simulazioaren exekuzioa gelditzen du.
- *Reset*: Botoi honek simulazioaren baldintza guztiak defektuzko hasierakoetara bueltatzen ditu.
- *Hurrengo pausua*: Botoi honen bidez denbora pausu bakar bat egiten da aurrera.



Irudia 3.5: Pendulu simplearen simulazioaren exekuzioa.

- *Erakutsi Deskripzioa*: Botoi honek simulazioaren deskribapen panela erakusten du, azalpenak panel honetan daude ere.

Parametroen aldaketa:

Ezkerretik hasita eskumara hau da karratu bakoitzak kontrolatzen duen parametroa:

- $l$  : Sokaren luzera aldatzen du.
- $m$  : Oszilatzen duen objektuaren masa aldatzen du.
- $r$  : Oszilatzen duen zirkuluaren erradioa aldatzen du.
- $\theta$  : Hasierako angeluaren balioa aldatzen du. Simulazioa martxan dagoenean erakusten du zein den bertikalarekiko angelua.
- $\omega$  : Hasierako abiadura angeluarra ezartzen du. Simulazioa martxan dagoenean erakusten du zein den abiadura angeluarra.

Hurrengo lau balioak ezin dira eskuz aldatu, lehen hirurak programak kalkulatzen ditu  $l$  luzera,  $\theta$  angelua eta  $\omega$  abiadura angeluarraren arabera (ikusi 3.21, 3.22 eta 3.23 ekuazioak).

- $v_x$  : Abiadura linealaren osagai horizontala.
- $v_y$  : Abiadura linealaren osagai bertikala.
- $v$  : Abiadura lineala.
- $t$  : Simulazioa abiarazi denetik pasatu den denbora.

Aldaketak egiteko nahi den balio numerikoa jarri behar da dagokion laukian eta *Enter* tekla sakatu. Baita ere posiblea da objektuaren hasierako posizioa aldatzea bere gainean klikatuz eta arrastratuz nahi den posizioraino.

Azkenik, eskuman dagoen barra bertikalak *zoom*-a aldatu egiten du. Osagai desberdinen gainean sagua segundo batzuk gelditu bada, horri buruzko informazio pixka bat duen laukizuzen bat agertzen da. Sagua fenomenoaren irudikapena gertatzen den tokiaren gainean uzten bada, simulazioan dauden geziei buruzko informazioa agertuko da. Geziak honako magnitudeei dagozkie:

- Gezi more horizontala abiadura lineal horizontala da,  $v_x$ .
- Gezi berde argi bertikala abiadura lineal bertikala da,  $v_y$ .
- Gezi magenta abiadura lineala da,  $\vec{v}$ .
- Gezi berde ilun bertikala grabitatea da.

Eskumako leihoan grafiko batzuk agertzen dira. Hauetan angeluaren eboluzioa denboraren menpean ( $\theta(t)$ ) ikusi daiteke eta baita ere energia irudikatzen da. Angeluaren eboluzioa beltzez irudikatzen da eta energia gorriaz. Angeluak gradutan irudikatuak dira.

### 3.2.4 Ariketak

1. **Angelu txikien hurbilketa:** Hasierako angelu txiki baterako ( $< 8^\circ$ ) eta objektua pausagunetik askatua izanda, ikusi  $\theta(t)$  grafikoa. Angelu txikien hurbilketan grafikoak kosinu itxura izan behar du eta periodoa ez da anplitudearen menpekoa izan behar, ikusi hau angelua pixka bat aldatuz (ez zara angelu txikien limitetik atera behar).
2. **Angelu txikien hurbilketaren limitea:** Hasierako angelua handituz joan grafikoa itxura kosinudala galdu eta periodoa anplitudearen menpekoa bihurtzen denerarte.
3. Ikusi alda daitezkeen parametroen eraginak. Periodoa zeren funtzioa da?

# Kapitulua 4

## Unibertsitate mailako simulazioak

Atal honetan unibertsitate mailarako egindako simulazioak aztertuko dira:  $N$  gorputzen problemaren adibide moduan Eguzki Sistema eta Atwood-en makina kulunkatzailea (*Swinging Atwood's machine*).

### 4.1 Eguzki Sistema

#### 4.1.1 Problemaren aurkezpena

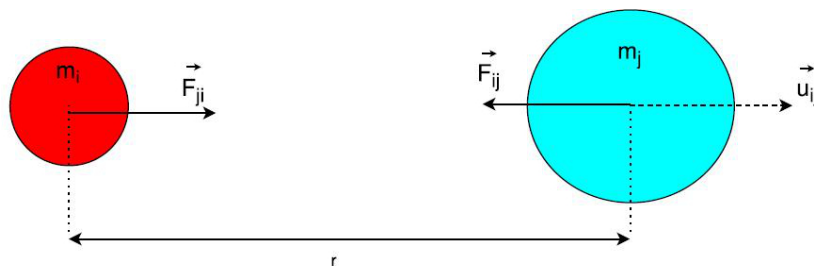
$N$  gorputzen probleman  $N$  gorputzek bata bestearekin duten elkarrakzioa aldi berean kontuan hartzen da. Adibide moduan Eguzki Sistema simulatu egin da, planetak (Merkurio, Artizarra, Lurra, Marte, Jupiter, Saturno, Urano eta Neptuno), Pluton eta Eguzkia kontuan hartuz Newton-en Grabitazioaren Lege Unibertsala erabiliz.

#### 4.1.2 Azalpen teorikoa

Jakina denez, Newton-en Grabitazioaren Lege Unibertsalak masa berdinen gorputzen arteko elkarrakzio grabitatorioa deskribatu egiten du. Hurrengo irudian (4.1 irudia) bi gorputzen arteko elkarrakzioaren eskema agertzen da.

$\vec{F}_{ij}$  da  $i$  gorputzak  $j$  gorputzaren gainean egiten duen indarra, eta indar grabitatorioaren kasuan, Newton-en Grabitazioaren Lege Unibertsalaren arabera, honako adierazpena du gorputzak puntualtzat hartzen badira:





Irudia 4.1: Bi gorputzen problemaren eskema.

$$\vec{F}_{ij} = -G \frac{m_i m_j}{|\vec{r}_j - \vec{r}_i|^2} \hat{u}_{ij} \quad (4.1)$$

non  $G$  den grabitazioaren konstante unibertuala ( $G = 6.67384 \times 10^{-11} \frac{Nm^2}{kg^2}$ ),  $m_i$  eta  $m_j$  dagokion gorputzaren masa,  $\vec{r}_i$  eta  $\vec{r}_j$  dagokion gorputzaren posizio bektorea erreferentzi sistema batekiko,  $|\vec{r}_j - \vec{r}_i|$  bi gorputzen arteko distantzia eta  $\hat{u}_{ij}$   $i$  gorputzetik  $j$  gorputzera doan bektore unitarioa. Ikus daiteke baita ere  $\vec{F}_{ij} = -\vec{F}_{ji}$  dela. Erabilgarriagoa izango da 4.1 ekuazioaren beste adierazpen hau:

$$\vec{F}_{ij} = -G \frac{m_i m_j}{|\vec{r}_j - \vec{r}_i|^3} (\vec{r}_j - \vec{r}_i) \quad (4.2)$$

N gorputzen sisteman, gorputz bakoitzak pairatzen duen indar totala izango da beste gorputz guztiek bere gainean sortzen duten indarren batura:

$$\vec{F}_i \text{ tot} = \sum_{\substack{j \\ i \neq j}} \vec{F}_{ij} \quad (4.3)$$

Kasu honetan ere higidura ekuazioak Newton-en bigarren legeak emandakoak izango dira.  $i$  gorputzarentzat, honakoa izango da:

$$m_i \ddot{\vec{r}}_i = \vec{F}_i \text{ tot} \quad (4.4)$$

Aurreko simulazioetako antzeko prozedura jarraituz hauek dira lortzen diren lehen ordenako ekuazio diferentzialak:

$$\dot{\vec{r}}_i = \vec{v}_i \quad (4.5)$$

$$\dot{\vec{v}}_i = \frac{\vec{F}_i \text{ tot}}{m_i} \quad (4.6)$$

Ekuazioak bektorialak direnez, guztira 6 ekuazio ditugu gorputz bakoitzeko.

### 4.1.3 Simulazioa

#### Simulazioa deskargatzeko:

<https://mega.co.nz/#!rRsG1SaZ!LkUpRu9DeZCINaY7rRlwbprAvi0QvKK-XDR1S08nQSI>

Simulazioan Eguzki Sistema simulatu egin da, zortzi planetak (Merkurio, Artizarra, Lurra, Marte, Jupiter, Saturno, Urano eta Neptuno), Pluton eta Eguzkia kontuan hartuz, 10 gorputzeko problema. Koordenatu sistemaren jatorri moduan masa zentroa aukeratu egin da.

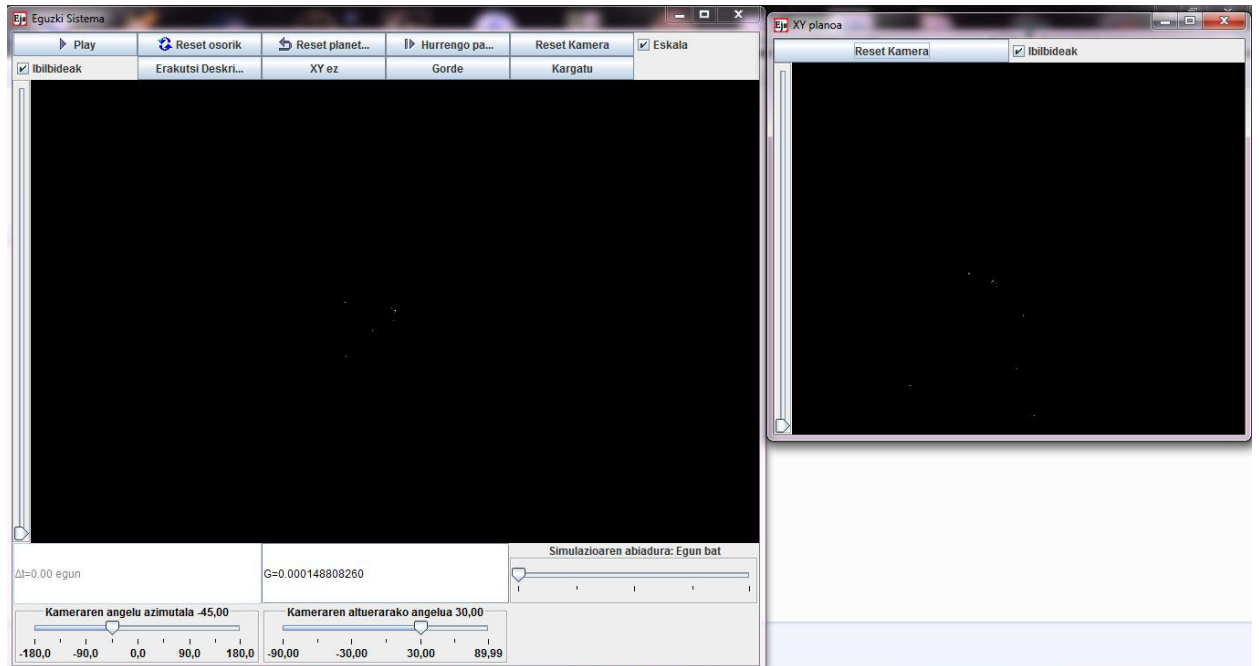
Erabiliko diren zenbaki handiak direla eta, unitate aldaketa bat egin da zenbakiak txikitzeko asmoz. Denbora unitate moduan egun bat erabili egin da, masa unitate moduan  $1kg' = 10^{30}kg$  eta distantzia unitate moduan unitate astronomikoa ( $1UA = 149.597.870.700m$ ). Aldaketa hauen ondorioz honela geratzen da grabitazioaren konstante unibertsala:  $G = 1.488082603 \times 10^{-4} \frac{UA^3}{kg'egun^2}$ . Honetaz gain, irudikapenean zerbait ikusi ahal izateko planeten eta Plutonen erradioen tamainak 1000 bider handitu dira eta Eguzkiaren erradioarena 100 aldiz handitu da.

Erabiliko diren objektuen masen eta erradioen balioak NASA-ren webgunetik (ikusi [7] erreferentzia) hartu egin dira eta posizio eta abiaduren hastapen balioak NASA-k efemerideak lortzeko eskaintzen duen *JPL's HORIZONS* sistema (ikusi [6] erreferentzia) erabiliz lortuak izan dira. Planeten hasierako posizio eta abiaduren balioak 1992ko apirilaren bikoak dira.

Simulazioan kontuan hartutako gorputz guztiak puntualtzat hartu egin dira ekuazioak ebazterako orduan. Beraz, 4.5, 4.6 ekuazioak 4.3 eta 4.2 ekuazioekin batera ebazteko Dormand-Prince8(53) metodoa aukeratu egin da kasu honetan ere.

Hurrengo irudian (4.2 irudian) ikusi egin daiteke simulazioa exekutatzean agertzen diren bi pantailak.

Ikus daitekeenez simulazioak bi pantaila ditu, ezkerreko leihoan 3 dimentsioko irudikapena egiten da eta eskumakoan 2 dimentsiokoa (XY plano). Ezkerrekoan baita ere daude simulazioa kontrolatzeko botoiak goialdean eta behealdean kameraren kokapena eta simulazioaren abiadura kontrolatzeko



Irudia 4.2: Eguzki Sistemaren simulazioaren pantaila.

aukerak.

#### **Ezkerreko leihoko botoiak:**

##### Goialdeko botoiak:

Hauek dira ezkerreko leihoko goialdeko botoien eginkizunak:

- *Play*: Simulazioa martxan jartzen du. Simulazio martxan badago *Play* botoiaren tokian *Stop* botoia egongo da eta azken honek simulazioaren exekuzioa gelditzen du.
- *Reset osorik*: Dena bueltatzen du hasierako balioetara, kamera barne.
- *Reset planetak*: Planeten posizio eta abiadurak hasierako balioetara bueltatzen ditu baina ez du kameraren posizio eta orientazioa aldatzen.
- *Hurrengo pausua*: Eboluzio pausu bakar bat aurreratzen du.
- *Reset Kamera*: Kameraren posizio eta orientazioa hasierakora bueltatzen ditu, planetak ukitu gabe.

- *Eskala*: Lauki hau aukeratuta egonez gero (tik bat agertzen da), has-tapen baldintzetan komentatutako eskala dago kontuan hartuta gorputz-tamainetan irudikatzeko orduan. Aukera hau kenduz gero, gorputz bakoitzaren tamaina modu baten handitzen da, nire ustez erosoagoa den era baten ikusteko planetak. Defektuz eskala mantendu egiten da.
- *Ibilbideak*: Lauki hau aukeratuta egonez gero gorputzen ibilbideak irudikatzen dira, ez badago aukeratuta ez dira irudikatzen. Defektuz marrazten dira.
- *Erakutsi Deskripzioa*: Deskripzio orria zabaltzen du.
- *XY ez*: XY plano irudikatzen duen leihoa ixten du eta *XY bai* botoiak leiho hau irekitzen du. *XY bai* botoia *XY ez* botoiaren tokian agertzen da.
- *Gorde*: Botoia sakatzen den momentuko planeten posizio, abiadurak eta pasatutako denbora osoa gordetzen ditu fitxategi baten.
- *Kargatu*: Azken aldiz gordetako planeten posizio, abiadurak eta pasatutako denbora osoa kargatzen ditu. Ez bada lehenago gorde, hasierako balioak kargatzen ditu.

#### Behealdeko botoiak:

Hauek dira toki honetan dauden bi lauki eta hiru scroll-en esanahia:

- $\Delta t$  : Zenbat egun pasa egin diren hasierako egunetik (1992-04-02)
- $G$  : Erabiltzen ari den grabitazioaren konstante unibertsalaren balioa, unitateak  $\frac{UA^3}{kg^1egun^2}$  izanik.
- *Simulazioaren abiadura*: Simulazioaren abiadura kontrolatzen du. Horretarako bista aktualizatu aurretik zenbat denbora pasatzen den zehazten du. Aukerak dira egun bat, aste bat, hilabete bat (30 egun), urte erdi (183 egun) edo urte bat (365 egun). Abiadura simulazioa exekutatzen dagoen bitartean ere alda daiteke.
- *Kameraren angelu azimutala*: Kameraren angelu azimutala finkatzen du marketan (45°-ko tarteetan daude kokatuta markak).
- *Kameraren altuerarako angelua*: Kameraren altuera zehazten du (30°-ko tarteetan). Altueraren angelua neurtzen hasten da XY planotik.

Hiru dimentsioko irudikapenetan kamera aldatzeko beste modu batzuk baita ere badaude:

- Saguarekin klikatu irudian eta mugitu sagua kameraren orientazioa aldatzeko.
- Teklatuko *shift* botoia zapaltzen ari den bitartean, saguaren ezkerreko botoia klikatuta izanda sagua mugitzen bada zoom egin daiteke.
- Teklatuko *Ctrl* botoia zapaltzen ari den bitartean, saguaren ezkerreko botoia klikatuta izanda sagua mugitzen bada irudia mugi egin daiteke.

Ezkerraldean dagoen scroll-ak leiho honetan zoom egiteko balio du.

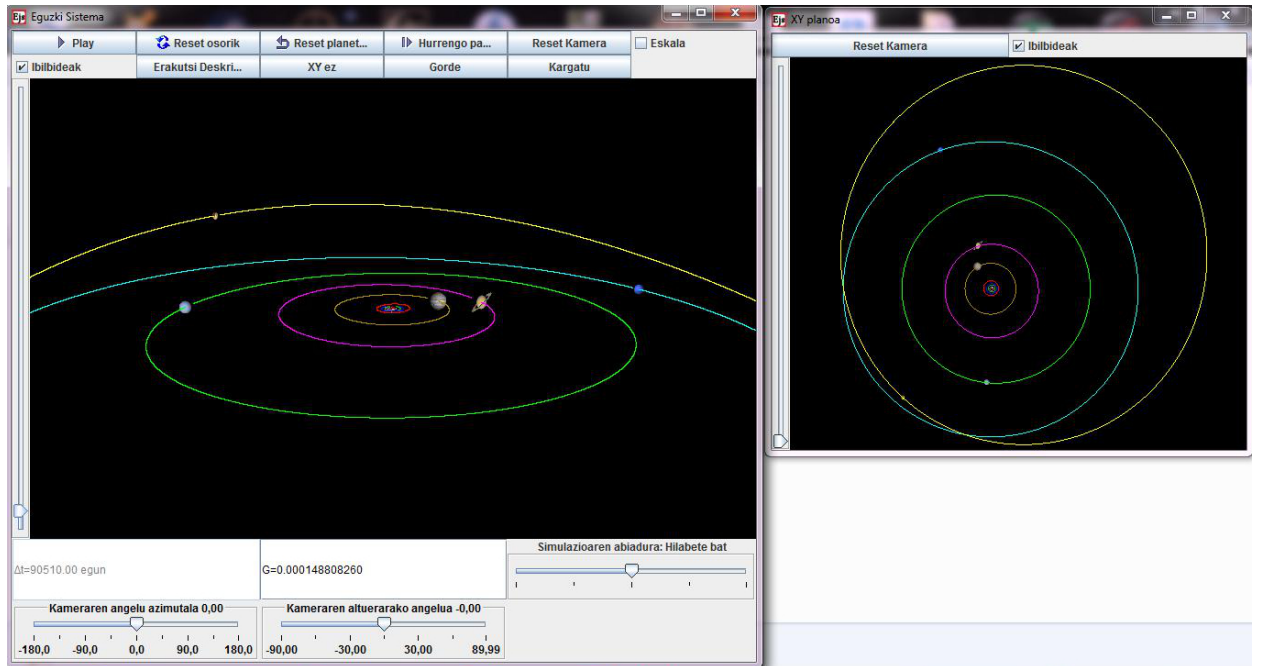
#### **Eskumako leihoko botoiak:**

Ezkerreko leihoan izen berdina duten botoien eginkizun berdina dute. Leiho honen ezkerraldean dagoen scroll-ak leiho honen zoom-a kontrolatzen du.

Hurrengo irudian (4.3 irudia) simulazioa martxan ikusi ahalko da, Plutonek lehenengo buelta bukatzen duenean dago hartuta irudia, simulazioaren abiadura hilabete batekoa aukeratu egin da, eskala aukera ez da aukeratu planetak hobeto ikusi ahal izateko, zoom pixka bat egin da eta kameraren orientazioa aldatuta dago.

Simulazioa exekutatzean *HasieraPlanetak* eta *OrainPlanetak* izeneko bi fitxategi sortuko dira simulazioaren exekutagarria gordeta dagoen karpetan. Bi fitxategietan gordeko dira aldagai berdinak:  $x, y, z, v_x, v_y, v_z$  eta  $t_{Osoa}$ . *HasieraPlanetak* fitxategiak balioko du planeten hasierako posizio eta abiadurara bueltatzeko kameraren orientazioa aldatu gabe *Reset Planetak* botoiaren bidez. *OrainPlanetak* fitxategiak ordea balioko du planeten posizioak, abiadurak eta igarotako denbora gordetzeko nahi den unean *Gorde* botoia erabiliz. *Kargatu* botoiaren bidez planeten gordetako azkeneko egoera kargatu ahalko da, inoiz ez bada bat gorde hasierako egoera kargatua izanik.

**OHARRA:** Simulazio honek problema bat dauka EJS programak erabiltzen duen libreriarekin 3 dimentsioetako trazan irudikapena dela eta. Denbora bat pasa ostean (ez da finkoa, batzuetan luzeagoa da, beste batzuetan laburragoa), 3 dimentsioko irudia biratzean simulazioa izozten da, konponbide bakar moduan simulazioa itxi eta berriz zabaltzea izanik. Arazoa konpontzeko asmotan EJS-ren egilearekin hitz egin dugu baita ere, badirudi sinkronizazio arazo bat dela eta igual momenturen baten zuzenduko du. Beraz, ez dut lortu problema konpontzea, lortu dudan gauza bakarra izan da simulazioa hasiera baten baino nahiko beranduago izoztea.



Irudia 4.3: Eguzki Sistemaren simulazioa.

#### 4.1.4 Ariketak

1. Simulazioa erabiliz neurtu gutxi gorabehera planeten periodo orbitala eta Keplerren Hirugarren Legea erabiliz ( $T^2 = \frac{4\pi^2}{GM_{\odot}} a^3$ ) kalkulatu planeten orbitaren ardatzerdi nagusia.
2. 4.3 irudiko bi dimentsioko proiektzioan, badirudi Pluton Neptunoren orbita zeharkatzen duela. Egia da?

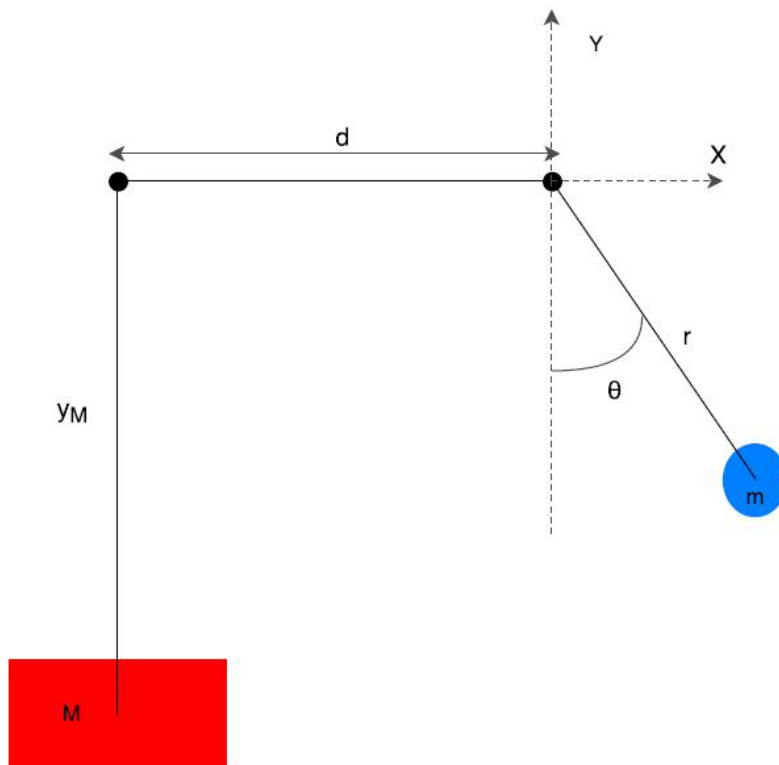
## 4.2 Atwood-en makina kulunkatzailea (*Swinging Atwood's machine*)

### 4.2.1 Problemaren aurkezpena

Atwood-en makina sistema mekaniko bat da, non bi masa lotuta dauden bi txirriketatik pasatzen den soka baten bidez. Atwood-en makina kulunkatzailearen kasuan masetako batek oszila dezake bertikalarekiko, bestea bakarrik bertikalki higi daitekeen bitartean. Simulazioan erabiliko den masa oszilakorraren forma zirkulu bat izango da eta bertikalean bakarrik higi daitekeenarena laukizuzen bat.

### 4.2.2 Azalpen teorikoa

Hurrengo irudian problemaren eskema bat ikus daiteke (4.4 irudia):



Irudia 4.4: Atwood-en makina kulunkatzailearen problemaren eskema.

Problema sinplifikatzeko, suposa dezagun txirrikak erradio gabekoak eta

leunak direla (marruskadurarik ez dagoela) eta soka masarik gabekoa eta luzaezina dela. Sokaren luzera osoari  $L$  deitzen bazaio,  $L$  konstante bat izango da eta baita 4.4 irudian agertzen den  $d$  txirriken arteko distantzia ere. Sistemak bi askatasun gradu dituzenez, bi koordenatu orokortu behar ditugu konfigurazioa (partikulen posizioa) lortzeko. Erreferentzi sistemaren jatorri moduan eskumako txirrika aukeratu egin da. Koordenatu orokortu moduan  $r$  eta  $\theta$  aldagaiak aukeratzen badira, 4.4 iruditik hauek dira  $m$  masaren posizioaren osagai kartesiarrak:

$$x_m = r \sin \theta \quad (4.7)$$

$$y_m = -r \cos \theta \quad (4.8)$$

Denborarekiko deribatuz abiaduraren osagaiak lor daitezke, eta hauetatik abiadura:

$$\dot{x}_m = \dot{r} \sin \theta + r \cos \theta \dot{\theta} \quad (4.9)$$

$$\dot{y}_m = -\dot{r} \cos \theta + r \sin \theta \dot{\theta} \quad (4.10)$$

$$v_m^2 = \dot{x}_m^2 + \dot{y}_m^2 = \dot{r}^2 + r^2 \dot{\theta}^2 \quad (4.11)$$

$M$  masaren posizioaren osagai bertikala lortzeko lotura ekuazioa erabiliz hauek dira posizioaren osagaiak:

$$x_M = -d \quad (4.12)$$

$$y_M = -(L - d - r) \quad (4.13)$$

$m$  masarekin bezala, denborarekiko deribatuz abiadurak lortzeko:

$$\dot{x}_M = 0 \quad (4.14)$$

$$\dot{y}_M = \dot{r} \quad (4.15)$$

$$v_M^2 = \dot{r}^2 \quad (4.16)$$

Beraz, sistemaren energi zinetikoa:

$$T = \frac{1}{2} M v_M^2 + \frac{1}{2} m v_m^2 = \frac{1}{2} (M + m) \dot{r}^2 + \frac{1}{2} m r^2 \dot{\theta}^2 \quad (4.17)$$

Eta energi potentziala:



$$V = Mgy_M + mgy_m = -Mg(L - d) - rg(-M + m \cos \theta) \quad (4.18)$$

Orduan, 4.17 eta 4.18 ekuazioak erabiliz, sistemaren lagrangearra:

$$\begin{aligned} L = T - V &= \frac{1}{2}(M + m)\dot{r}^2 + \frac{1}{2}mr^2\dot{\theta}^2 + Mg(L - d) + rg(-M + m \cos \theta) = \\ &= \frac{1}{2}(M + m)\dot{r}^2 + \frac{1}{2}mr^2\dot{\theta}^2 + rg(-M + m \cos \theta) \end{aligned} \quad (4.19)$$

non azken berdintzan lagrangearreko osagai konstantea kendu egin den ez duelako eraginik. Higidura ekuazioak lortzeko formalismo Hamiltondarra erabiliko da formalismo honetan zuzenean agertzen direlako higidura ekuazioetan lehen ordenako ekuazio diferentzialak.

Karrerako bigarren mailan ikusi egin zen bezala hamiltondarra  $\vec{q}$  koordenatu orokortu eta  $\vec{p}$  momentu kanoniko konjokatuen funtzio idatzi behar da:

$$H(t, \vec{q}, \vec{p}) = \sum_i p_i \dot{q}_i(t, \vec{q}, \vec{p}) - L(t, \vec{q}, \dot{\vec{q}}(t, \vec{q}, \vec{p})) \quad (4.20)$$

Definizioz momentu kanoniko konjokatuak  $p_i = \frac{\partial L}{\partial \dot{q}_i}$  moduan kalkulatu dira. Beraz, Atwood-en makina kulunkatzailearen kasuan hauek dira dauden bi momentu kanoniko konjokatuak:

$$p_r = \frac{\partial L}{\partial \dot{r}} = (M + m)\dot{r} \quad \Rightarrow \quad \dot{r} = \frac{p_r}{M + m} \quad (4.21)$$

$$p_\theta = \frac{\partial L}{\partial \dot{\theta}} = mr^2\dot{\theta} \quad \Rightarrow \quad \dot{\theta} = \frac{p_\theta}{mr^2} \quad (4.22)$$

Ondorioz, 4.19 lagrangearra  $t$  denbora,  $r$  eta  $\theta$  koordenatu orokortu eta 4.21 eta 4.22 ekuazioek emandako  $p_r$  eta  $p_\theta$  momentu kanoniko konjokatuen funtzio:

$$L(t, \vec{q}, \dot{\vec{q}}(t, \vec{q}, \vec{p})) = \frac{1}{2} \frac{p_r^2}{M + m} + \frac{1}{2} \frac{p_\theta^2}{mr^2} + rg(-M + m \cos \theta) \quad (4.23)$$

Beraz, sistemaren hamiltondarra honakoa da formalismo honetan:

$$H = p_r \dot{r} + p_\theta \dot{\theta} - L = \frac{1}{2} \frac{p_r^2}{M + m} + \frac{1}{2} \frac{p_\theta^2}{mr^2} - rg(-M + m \cos \theta) \quad (4.24)$$

Formalismo hamiltondarrean higidura ekuazioak  $\dot{q}_i = \frac{\partial H}{\partial p_i}$  eta  $\dot{p}_i = -\frac{\partial H}{\partial q_i}$  dira. Orduan, adierazpen horiek erabiliz, hauek dira Atwood-en makina kulunkatzailearen lau higidura ekuazioak:

$$\dot{r} = \frac{\partial H}{\partial p_r} = \frac{p_r}{M+m} \quad (4.25)$$

$$\dot{\theta} = \frac{\partial H}{\partial p_\theta} = \frac{p_\theta}{mr^2} \quad (4.26)$$

$$\dot{p}_r = -\frac{\partial H}{\partial r} = \frac{p_\theta^2}{mr^3} + g(-M + m \cos \theta) \quad (4.27)$$

$$\dot{p}_\theta = -\frac{\partial H}{\partial \theta} = -mgr \sin \theta \quad (4.28)$$

Sistema hau masen arteko zatiduraren oso menpekoa da:  $\mu = \frac{M}{m}$

### 4.2.3 Simulazioa

#### Simulazioa deskargatzeko:

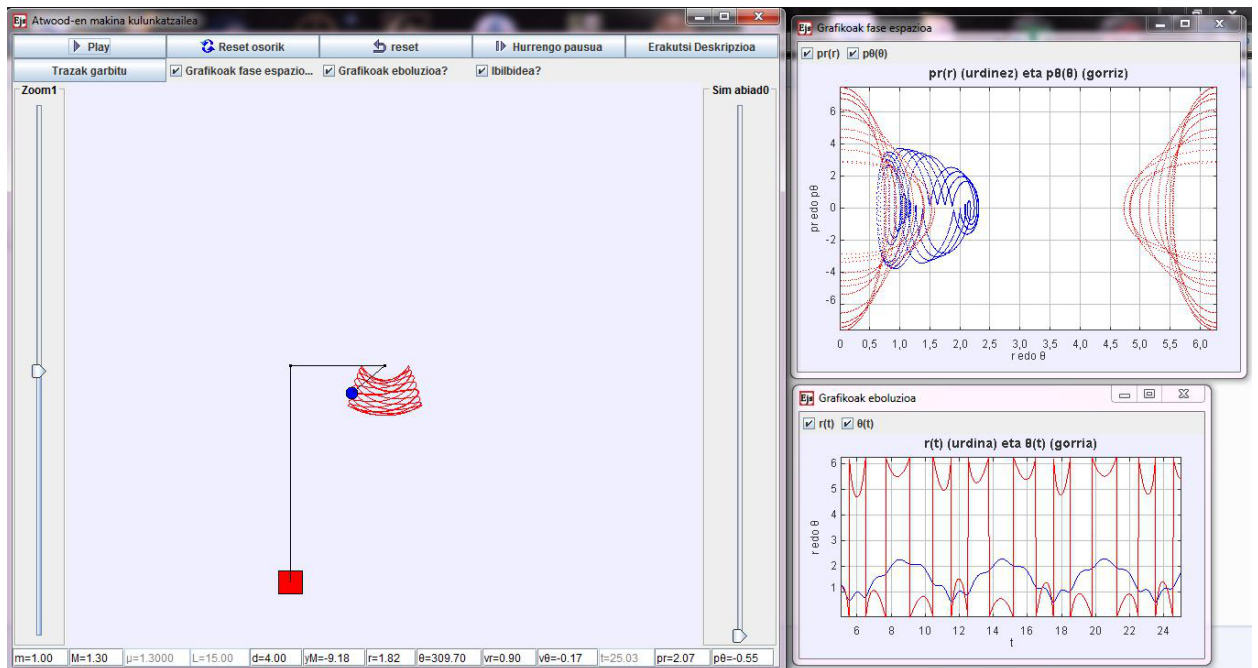
<https://mega.co.nz/#!jI11DSoA!FEoFbFepbqio7-g0X8rcrJjrzvKTsQxTt1u8Nqo0oII>

Programak 4.25 - 4.28 higidura ekuazioak ebazten ditu Dormand-Prince8(53) metodoa erabiliz hasierako egoera batetik hasita sistemaren konfigurazioa ezagutzeko denbora pausu bakoitzean. Defektuzko hasierako egoera da masa oszilakorra geldu egotea bertikalarekiko  $90.0^\circ$ -ko angeluarekin eta eskumako txirrikatik distantzia erradiala 1 izatea, bertikalki bakarrik higi daitekeen masaren posizioa -10 izatea eta masen arteko ratioa  $\mu = 1.3$ . Hurrengo irudian (4.5 irudia) ikus daiteke simulazioaren pantaila, irudi honetan simulazioa abiarazi egin da komentatutako hasierako egoeratik.

Simulazioak hiru pantaila dauzka. Aurreko simulazioetan bezala, ezkerreko pantailaren goialdean simulazioa kontrolatzeko botoiak daude, eta behealdeko laukietan sistemaren parametroak eskuz aldatzen dira. Ezkerreko scroll bertikala zoom-a da eta eskumakoak simulazioaren abiadura kontrolatzen du. Beste bi pantailatan grafiko batzuk irudikatu egiten dira. Hauek dira ezkerreko leihoko botoien eta laukien eginkizunak:

#### Goialdeko botoiak:

- *Play*: Simulazioa martxan jartzen du. Simulazio martxan badago *Play* botoiaren tokian *Stop* botoia egongo da eta azken honek simulazioaren exekuzioa gelditzen du.



Irudia 4.5: Atwood-en makina kulunkatzailearen simulazioaren pantaila.

- *Reset Osorik*: Simulazioaren baldintza guztiak defektuzko hasierakoe-tara bueltatzen ditu.
- *Reset*: Erabili diren azken hastapen baldintzetara bueltatzen du simu-lazioa.
- *Hurrengo pausua*: Denbora pausu bakar bat egiten da aurrera.
- *Erakutsi Deskripzioa*: Simulazioaren deskribapen orria erakusten du.
- *Trazak garbitu*: Traza guztiak ezabatzen ditu.
- *Grafikoak fase espazioa?*: Fase espazioak ( $p_r(r)$  eta  $p_\theta(\theta)$ ) irudikatzen diren grafikoaren leihoa erakustea edo ezkutatzea kontrolatzen du.
- *Grafikoak eboluzioa?*: Eboluzioak ( $r(t)$  eta  $\theta(t)$ ) irudikatzen diren gra-fikoaren leihoa erakustea edo ezkutatzea kontrolatzen du.
- *Ibilbidea?*: Leiho nagusian masa oszilakorraren ibilbidea marraztea edo ez kontrolatzen du.

Behealdeko laukiak:

- $m$  : Gorputz oszilakorraren masa (gorputz urdin zirkularra) aldatzen du.
- $M$  : Bertikalki bakarrik highi daitekeen gorputzaren masa (gorutz gorri laukizuzena) aldatzen du.
- $\mu$  : Masa ratioa. Balio hau ezin da aldatu hemen, gorputzen masak aldatuz aldatzen da.
- $L$  : Sokaren luzera osoa. Balio hau ezin da aldatu hemen. Beste  $d$ ,  $y_M$  eta  $r$  parametroak aldatuz alda daiteke.
- $d$  : Bi txirriken arteko distantzia.
- $y_M$  : M masaren hasierako posizio bertikala.
- $r$  : Eskumako txirrikatik m masara dagoen distantzia erradiala.
- $\theta$  : m masak bertikalarekin osatzen duen angelua.
- $v_r$  : m masaren abiadura erradiala.
- $v_\theta$  : m masaren abiadura angeluarra
- $t$  : Simulazioa abiarazi denetik pasatutako denbora.
- $p_r$  :  $p_r$  momentu kanoniko konjokatuaren balioa.
- $p_\theta$  :  $p_\theta$  momentu kanoniko konjokatuaren balioa.

Grafikoen leihoetan agertzen diren laukiek kontrolatzen dute ze grafikoz marrazten diren:

- $v_r(r)$  :  $v_r$   $r$ -ren funtzio irudikatzea edo ez kontrolatzen du.
- $v_\theta(\theta)$  :  $v_\theta$   $\theta$ -ren funtzio irudikatzea edo ez kontrolatzen du.
- $r(t)$  :  $r$   $t$ -ren funtzio irudikatzea edo ez kontrolatzen du.
- $\theta(t)$  :  $\theta$   $t$ -ren funtzio irudikatzea edo ez kontrolatzen du.

Simulazioa exekutatzean *HasieraAtwood* izeneko fitxategi bat sortuko da simulazioaren exekutagarria gordeta dagoen karpetan. Hemen gordeko da sistemaren hasierako egoera (justu simulazioa martxan jarri aurretik duguna), defektuzkoa edo aldatua aldaketak egin badira. Gordeko diren aldagaiak  $m$ ,  $M$ ,  $d$ ,  $y_M$ ,  $r$ ,  $\theta$ ,  $v_r$ ,  $v_\theta$ ,  $p_r$  eta  $p_\theta$  dira. Egoera honetara itzuli ahalko da *Reset*

botoia erabiliz.

Masetakoren batek txirrika jotzen badu simulazioa geldituko da eta laugarren leiho bat irekiko da esanez zein masak jo duen txirrika.

#### 4.2.4 Ariketak

Sistema honek hainbat ibilbide mota desberdin izan ditzake (ikusi [3] erreferentzia). Ibilbideek masa ratioarekiko menpekotasun handia dute.

- *Ibilbide mugatua*: Ibilbidea mugatua dela esaten da baldin eta  $r$  beti bada balio maximo bat baino txikiagoa. Atwood-en makina kulunkatzailearen mugimendua mugatua da  $M > m$  bada.
- *Ibilbide periodikoa*:  $\tau$  denbora bat pasa ostean ibilbidea errepikatzen bada.
- *Ibilbide singularra*: Momenturen baten  $r$  0-ra badoa ibilbidea singularra dela esaten da, era sinpleena  $r(0) = 0$  egitea da.
- *Bukatzen den ibilbide singularra*: Aurreko kasuko azpikasu bat da,  $m$  txirrikaren kontra jotzen badu,  $M$ -k bat-batean norabidea aldatu behar du.

1. **Ibilbide desberdinen azterketa**: Ibilbide desberdinak aztertzeko hurrengo balioak proba ditzakezu.

- (a) Ibilbide mugatu ez-singularrak aztertzeko askatu pausagunetik  $m$  masa hasieran  $\theta = 90^\circ$  eta  $r = 1$  izanik. Aldatuz joan  $\mu$ -ren balioa ibilbide desberdinak ikusteko (adibide moduan 1.4, 2.2, 3.0,...).
- (b) Ibilbide periodikoak ikusteko askatu aurreko kasuko hasierako baldintzetatik adibidez honako masa ratioaren balioekin: 1.1185, 1.555, 1.665, 2.394,... Probatu beste balio batzuk ea besteren bat lortzen duzun (igual ibilbide mugatu ez-singularrak aztertzean topatu duzu dagoeneko periodiko bat). Ikus daiteke ibilbide periodikoak lortzea zailagoa dela ibilbide mugatu ez-singularrak baino.
- (c) Ibilbide singularrak aztertzeko bota masa horizontalki zentrotik ( $r = 0$ ) abiadura erradial batekin. Adibidez saiatu  $v_r = 4$ -rekin eta  $\mu$ -ren balio hauetarako: 1.1 (honekin ere aldatu  $y_M = -20.0$ -ra adibidez), 1.6, 2.3,...

- (d) Bukatzen diren ibilbide singularrak aztertzeke aurreko kasuko hasierako egoeratik bota  $m$  masa honako masa ratioekin esate baterako: 3.0, 1.182, 1.299, 1.594, 2.827,...
2. Aldatu hasierako parametroak nahi diren moduan eta aztertu zer nolako ibilbidea ateratzen den.

OHARRAK:

- Denbora pasa ahala simulazioan ibilbide periodikoen periodikotasuna galtzen da, uste dut hau izan daitekeela zenbakizko erroreengatik.
- $r = 0$ -tik askatzea ezin da higidura ekuazioetan  $r$ -rekin zatidurak daudelako, orduan erabiltzaileak eskuz jartzean  $r = 0$  programak  $r = 0.001$  egingo du.
- Ariketetan jarri egin diren bukatzen diren ibilbide singularren kasuko masa ratio gehienetarako simulazioan ez da agertuko masak txirrika jo duela esaten duen leihoa (masak txirrikaren kontra “errebotatzen” duela ikusten da).  $\mu = 3.0$  kasuan bai zabaltzen da leihoa, esanez  $m$  masak txirrika jo duela.

## Kapitulua 5

# Ondorioak

Lan honen garapenetik atera egin dudan lehen ondorioa izan da Easy Java Simulations programa oso erabilgarria egin izan zaidala simulazio interaktiboak era erraz baten sortzeko. Lanaren helburua fisika irakasteko simulazioak garatzea zenez, simulazioak interaktiboak izatea garrantzitsua da, horrela ereduizatzen ari den fenomenoan parte hartzen duten parametroen eragina ikus daiteke hauen balioak aldatzean simulazioan duten eragina aztertuz.

Simulazio bakoitzetik honakoa azpimarratuko nuke:

- *Tiro parabolikoa*: Hau izan zen egin nuen lehena, horregatik, simulazio honetan izan nituen arazo asko lotuta zeuden programaren erabileran berria nintzenarekin. Simulazio honen beste bertsio baten jaurtigaia markaren gainean erori egin den edo ez esaten du programak, baina azkenean ez zitzaidan asko gustatu nola geratu zitzaidan eta simulazioaren azkeneko bertsiorako kentzea erabaki egin nuen. Baita ere hedatu egin nuen simulazio hau hiru dimentsioetara, baina argiagoa iruditzen zitzaidanez bi dimentsiokoa hau da txostenean jarri egin dudana.
- *Pendulu simplea*: Simulazio honetan arazoak eman didana izan da nola lortu pantailaren behealdean ikusten den  $\theta$  angeluaren zenbakia nahi nuen tartean agertzea, hau da  $(0^\circ, 360^\circ)$  tartean egotea.
- *Eguzki Sistema*: Hau izan da gehien egiteko gogoia nuen simulazioa, proiektua aukeratu nuenetik hau egitea pentsatuta nuelako. Simulazio honek buruhauste nahiko eman dizkit, batez ere 4.1.3 ataleko oharrean komentatutako trazaren arazoa dela eta simulazioa izoztea.
- *Atwood-en makina kulunkatzailea*: Simulazio honetan arazoak eman didana izan da bukatzen diren ibilbide singularren kasuko azterketa,

---

ez dudalako lortu [3] erreferentzian agertzen diren kasu honi dagozkion masa ratio guztiekin txirrika jo dela esaten duen leihoa irekitzea. Suposatzen dut hau izan daitekeela masa justu jatorritik bota ezin izatearen edo zenbakizko kalkuluaren ondorioz m masak ez duela betetzen justu txirrikaren kontra talka egon dela esan nahi duen jarritako baldintza.

Azkenik, egindako simulazioen erabilgarritasun didaktikoari buruz, nire ustez lehenengo biak didaktikoagoak dira azkeneko biak baino. Baita komentatu nahiko nuke ikusi egin dudala ariketak pentsatzea ez dela gauza erreza.



# Bibliografia

- [1] Juan Mari Aguirregabiria MEKANIKA KLASIKOA. Euskal Herriko Unibertsitatea, 2012.
- [2] Francisco Esquembre CREACIÓN DE SIMULACIONES INTERACTIVAS EN JAVA. Madrid: Peason Educación, S.A., 2005.
- [3] Nicholas B. Tufillaro, Tylear A. Abbott eta David J. Griffiths. “Swinging Atwood’s Machine”. AMERICAN JOURNAL OF PHYSICS 1984ko urria, 52. alea, 895-903 orriak: <http://dx.doi.org/10.1119/1.13791>
- [4] Hugh D. Young y Roger A. Freedman FÍSICA UNIVERSITARIA VOLUMEN 1 Decimosegunda edición. México: Pearson Educación, 2009.
- [5] Easy Java Simulations Wiki: <http://www.um.es/fem/EjsWiki/>. Kontsulta: Hainbat egunetan proiektuaren garapenean zehar.
- [6] JPL HORIZONS System: <http://ssd.jpl.nasa.gov/horizons.cgi>. Kontsulta: 2015-01-28
- [7] NASA Space Science Data Coordinated Archive: <http://nssdc.gsfc.nasa.gov/>. Kontsulta: 2014-12-24