Títol: Laboratori de física a la "web"

Alumne: David Pérez Alberto

Director/Ponent: Jordi Martí Rabassa
Departament: FEN
Data:28/06/2011

DADES DEL PROJECTE

Títol del Projecte: Laboratori de física a la “web”

Nombre del estudiant: David Pérez Alberto

Titulació: Enginyeria Tècnica en Informàtica de Gestió

Crèdits: 22,5

Director/Ponent: Jordi Martí Rabassa

Departament: FEN

MEMBRES DEL TRIBUNAL *(nom i signatura)*

President: Joaquim Valls Ribas

Vocal: Josep Carmona Vargas

Secretari: Jordi Martí Rabassa

QUALIFICACIÓ

Qualificació numèrica:

Qualificació descriptiva:

Data: 28/06/2011

Índex

0 Introducció i descripció general del projecte 8
0.1 Model General 9
0.2 Model Harmònic: 9
0.3 Model Xocs: 9
1 Estudi comparatiu de laboratoris virtuals 10
2 Introducció a la física dels models implementats 16
2.1 Simulació física. Mètodes numèrics: 16
2.1.1 Avaluació de les equacions diferencials 16
2.2 Mètode d'Euler 17
2.2.1 Descripció geomètrica informal 17
2.2.2 Derivació 18
2.2.3 Error 18
2.3 Mètode Runge-Kutta 19
2.4 Moviment rectilini uniforme (MRU): 21
2.5 Moviment rectilini uniformement accelerat (MRUA) 22
2.5.1 Equació de la posició: 22
2.5.2 Equació de la velocitat: 22
2.6 Moviment harmònic simple (MHS): 23
2.6.1 Magnituds i caracterització del MHS 23
2.6.2 Oscil·lador lliure: 24
2.6.3 Oscil·lador esmorteït: 25
2.6.4 Oscil·lador forçat: 27
2.7 Camp elèctric 36
2.7.1 Definició electrostàtica 36
2.7.2 Moviment en un camp elèctric 37
2.8 Camp magnètic 38
2.8.1 Camp solenoïdal 38
2.8.2 Vector axial 39
2.8.3 Moviment en un camp magnètic 40
2.8.4 Moviment circular uniforme 41
2.8.5 Angle i velocitat angular 41
2.8.6 Vector de la posició 42

2.8.7 Vector de velocitat 42
2.8.8 Vector d’Acceleració 43
2.9 Moviment en camps elèctric i magnètic creuats 44
2.9.1 Casos particulars 47
2.10 Xocs de partícules 50
2.10.1 Xocs elàstics 50
2.10.2 Xocs inelàstics 50
2.10.3 Coeficient de restitució 50
2.10.4 Xocs frontals de partícules 52
2.10.5 Xocs bidimensionals 54
Cas particular 55
3 Especificació 56
3.1 Especificació casos d'ús 57
3.1.1 Diagrames de casos d’us 57
3.2 Especificació estats 67
3.2.1 Diagrama d’estats 67
3.3 Model de disseny 68
3.3.1 Patró de disseny principal: Model vista controlador 68
3.3.2 Descripció del patró 68
3.4 Model de desenvolupament 71
3.4.1 Prototip : desenvolupament evolutiu 71
3.5 Diagrama de classes: 73
3.5.1 Model 75
3.5.2 Controlador 79
3.5.3 Vistes 81
4 Disseny de l’ interfície 82
4.1 Estructura general de la interfície 83
5 Eines utilitzades per realitzar el projecte 85
5.1 JScience 85
5.1.1 Exemple d’utilització: 86
5.2 Java Excel API 89
5.2.1 Exemple d’utilització: 90
5.3 JNLP 92

5.3.1 Exemple d'utilització: 92
5.4 Serialització 94
5.4.1 Exemple d'utilització: 94
5.5 IDEs utilitzats 96
6 Planificació 97
6.1 Definició de tasques 97
6.2 Assignació de tasques 99
7 Valoració econòmica 100
7.1 Cost de personal 100
7.2 Cost de software utilitzat 100
7.3 Cost de hardware per el desplegament 101
8 Conclusions 102
9 Propostes de millora 103
10 Agraïments 104
11 Bibliografia 105
12 Annexos 106
Mètodes de Runge-Kutta de segon ordre 107
Mètodes de Runge-Kutta de tercer ordre 108
Mètodes de Runge-Kutta de quart ordre 108
13 Referències 110

Taula d'il·lustracions

Il·lustració 1: Vista pàgina web myphysicalab 10
Il·lustració 2: Vista pàgina web myphysicalab 11
Il·lustració 3: Vista pàgina web Walter-Fendt 12
Il·lustració 4: Vista pàgina web PhET 13
Il·lustració 5: Vista pàgina web EjS 14
Il·lustració 6: Mètode d'Euler 17
Il·lustració 7: Oscil·lador lliure 24
Il·lustració 8: Oscil·lador esmorteït 25
Il·lustració 9: Equació amplitud oscil·lació esmorteïda. 27
Il·lustració 10: Oscil·lador forçat. 27
Il·lustració 11: Amplitud màxima en un període 30
Il·lustració 12: Amplitud en dos períodes de ressonància 30
Il·lustració 13: Amplitud creixent per ressonància 31
Il·lustració 14: Diagrama vectorial del MHS 32
Il·lustració 15 Amplitud del oscil·lador fins arribar a l'estabilitat. 33

Il·lustració 16 Energia del oscil·lador en funció del temps..... 33
Il·lustració 17 Amplitud creixent de l'oscil·lador,..... 34
Il·lustració 18 Espai de fases x-v (posició -velocitat). 35
Il·lustració 19 Regla de la mà dreta. 40
Il·lustració 20 Definició vectorial de forces 44
Il·lustració 21 Moviment d'una partícula amb camp elèctric i magnètic..... 47
Il·lustració 22 Moviment d'una partícula amb camp magnètic i camp elèctric nul. 48
Il·lustració 23 Moviment sobre camp elèctric i magnètic però sense $V0y$ 48
Il·lustració 24 Cicloide creat al no tenir velocitat inicial..... 49
Il·lustració 25 Estats abans i després d'un xoc entre dos partícules. 51
Il·lustració 26 Comportament masses en un xoc de partícules, ... 52
Il·lustració 27 Comportament velocitats en un xoc de partícules..... 53
Il·lustració 28 Comportament xoc bidimensional de partícules. 54
Il·lustració 29 Cas d'ús – Menú principal..... 57
Il·lustració 30 Cas d'ús –Experiment General..... 59
Il·lustració 31 Cas d'ús –Experiment Harmònic..... 61
Il·lustració 32 Cas d'ús –Experiment Xocs 64
Il·lustració 33 Diagrama d'estats 67
Il·lustració 34 Patró de disseny MVC..... 68
Il·lustració 35 MVC amb l'usuari 69
Il·lustració 36 Model de prototipus..... 71
Il·lustració 37 Diagrama de classes 74
Il·lustració 38 Classe Partícula 75
Il·lustració 39 Classe Molla 75
Il·lustració 40 Classe ModelExperimentGeneral 75
Il·lustració 41 Classe ModelExperimentXocs 76
Il·lustració 42 Classe ModelExperimentXocs 76
Il·lustració 43 Classe Vector 77
Il·lustració 44 Classe BoolAmount..... 77
Il·lustració 45 Classe DosStrings 77
Il·lustració 46 Classe Bool2Double 77
Il·lustració 47 Classes Exportació a Excel 78
Il·lustració 48 Controlador Experiment General 79
Il·lustració 49 Controlador ExperimentMolla..... 80
Il·lustració 50 Controlador Experiment Xocs..... 80
Il·lustració 51 Controlador exportació i serialització de dades 80
Il·lustració 52 Classes de la capa de presentacions..... 81
Il·lustració 53 Aspecte de l'aplicació 82

0 Introducció i descripció general del projecte

El projecte que a continuació es descriu sorgeix per cobrir la necessitat de crear una eina de representació de experiències físiques, tal i com es faria a un laboratori convencional, però de manera virtual. A la web del departament de Física i Enginyeria Nuclear[i], hi ha una col·lecció d’aplicacions Java (applet) sobre diferents conceptes de física bàsica, són exemples d’ altres projectes realitzats per estudiants sobre temes molt concrets. El projecte es diferenciarà sobre aquests altres perquè s’assembla més a una eina de simulació i posterior anàlisi de dades sobre els experiments realitzats.

Els objectius del projecte són: implementar una eina de simulació de models físics en dues dimensions, que permetin a l’usuari visualitzar/analitzar amb les dades obtingudes, el comportament d’aquest models en les diferents configuracions possibles escollides per l’usuari. Aquesta eina esta orientada per a la docència, ja que permet recrear situacions que l’estudiant està aprenent d’una manera visual i interactiva, facilitant l’aprenentatge del comportament físic d’aquestes experiències.

Per acotar l’abast del projecte es van definir uns models sobre els quals es generarien experiments, però a l’aplicació es podrien implementar altres models als que es van definir, a més podrien aprofitar els components comuns per el seu funcionament, reduint el cost de desenvolupament de noves funcionalitats.

Aquets són els tres models que s’han implementat:

0.1 Model General:

Partícula lliure de massa, pes, radi i càrrega elèctrica variable, sotmesa a la gravetat variable, camp magnètic variable, però perpendicular al pla del moviment (eix z), i també camp elèctric variable. També es podrà modificar les condicions inicials del moviment de la partícula: posició inicial, velocitat inicial i acceleració. L’espai de l’experiment podrà tenir o no límits, permeten rebots amb les parets, també es podrà variar el tipus de xoc depenent del coeficient de restitució de la força quan es produeixen col·lisions.

0.2 Model Harmònic:

Partícula fixada a una molla (oscil·lador harmònic) de massa i pes variable. La molla de longitud, posició, elongació inicial i constant elàstica variable, podrà estar sotmesa a una força externa amb una freqüència angular variable. També es podrà variar el coeficient de esmorteïment del moviment de la molla.

0.3 Model Xocs:

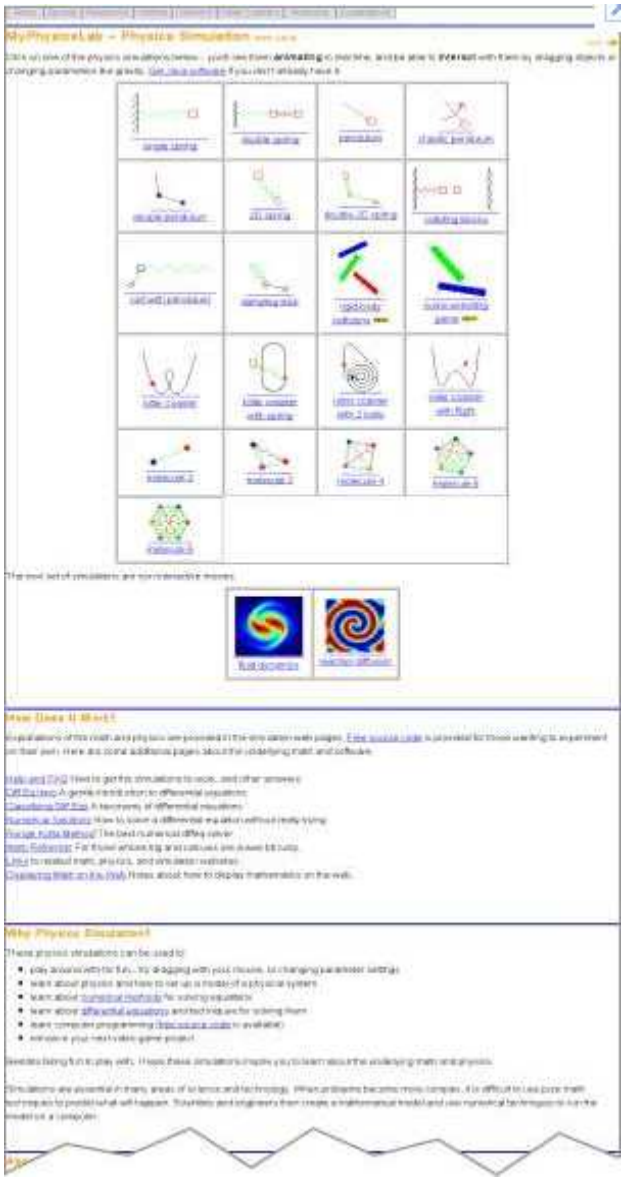
Dues partícules lliures de pes, massa i radi variables. També es podrà modificar les condicions inicials del moviment de la partícula: posició inicial, velocitat inicial i acceleració. L’espai del model estarà limitat ja que afavorirà que es produeixin col·lisions entre les partícules presents a l’experiment.

Sobre aquests models s’ha implementat una eina que permet realitzar la exportació de dades que es produeixen a les simulacions en curs a un full de càlcul (.xls). L’usuari podrà escollir les variables que s’exportaran, per que es pugui realitzar un anàlisi posterior de les dades.

També per facilitar l’ utilització del programa s’ha implementat la possibilitat de carregar i desar la configuració del experiment, per tant es podrà desar i carregar les propietats del experiment per poder-ho reproduir en qualsevol moment, facilitant així l’ utilització del programa.

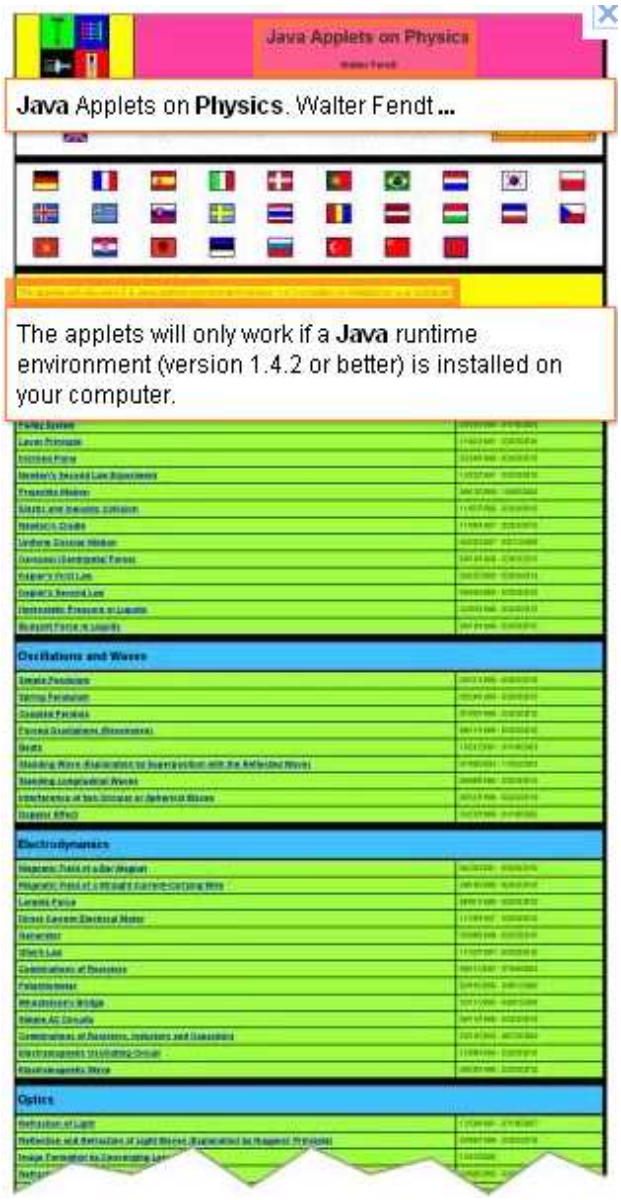
1 Estudi comparatiu de laboratoris virtuals

A continuació s'analitzen quatre laboratoris virtuals, cadascun d'ells té alguna característica en particular, ja que existeixen una pila de webs que tracten sobre aquest tema. El que s'ha intentat es d'escollir les millors i més completes per extreure els punts forts que pot tenir un laboratori web. També s'ha afegit una cinquena pagina web que tracta d'una eina molt utilitzada en simulacions de física.

Pàgina web	Anàlisis
http://www.myphysicslab.com/  **Il·lustració 1:** Vista pàgina web myphysicalab	Pàgina web que conté diversos experiments, sobre tot d'experiències de cinemàtica. Aquestes estan representades gràficament de manera que semblen molt reals. Aquesta web destaca per la qualitat gràfica dels experiments, el nivell d'interacció i també per la possibilitat de veure les propietats del experiment representats en gràfics en temps real. També com a contrapartida, en algunes experiències s'assemblen més a jocs que no a simulacions físiques, ja que no aporten coneixements físics, únicament son divertits . Punts a destacar: -Permet interactuar amb el models. -Genera gràfics en temps real. -Molt realista.

Pàgina web	Anàlisis
http://www.phy.ntnu.edu.tw/ntnujava/ **Il·lustració 2:** Vista pàgina web myphysicalab	Un altre laboratori però aquest esta pensat com un fòrum, on els membres poden afegir experiències, classificades en funció de l'àmbit físic i també en funció de com han estat creades. El tret a destacar d'aquesta web es la quantitat i varietat d'experiments que s'inclouen, aquestes estan classificades en mecànica, cinemàtica, òptica etc... Té dos apartats de simulacions, un de simulacions fetes amb Java i un altre que estan fetes amb EJS[1] (En pàgines posteriors s'explica que es EJS amb detall). Cal destacar algunes simulacions en 3D sobre partícules sotmeses a camp elèctric i magnètic. Punts a destacar: - Gran varietat de simulacions – tipus. - Qualitat dels experiments físics.

[1] Easy Java Simulations és una eina de programari dissenyada per a la creació de simulacions discretes per computador. El que fa EJS diferent de la majoria dels altres productes és que EJS no ha estat dissenyat per fer la vida més senzilla als programadors professionals, sinó que ha estat concebut per professors de ciències per ser usat per professors i estudiants de ciències. Això és, per a persones que estan més interessades en el contingut de la simulació, en el fenomen mateix que se simula, que en els aspectes tècnics necessaris per construir la simulació.

Pàgina web	Anàlisis
http://www.walter-fendt.de/ph14e/  **Il·lustració 3:** Vista pàgina web Walter-Fendt	Pàgina web senzilla, però que conté molts experiments. El nivell del experiments va des de molt senzills i per (estudiants de primària) fins a experiments més complexos (per batxillerat o universitat). L'estètica de les simulacions és de dibuixos animats, no són molt acadèmics. Un punt important es que la web i els experiments estan traduïts a uns 30 idiomes diferents aportant un valor afegit ja que aquests poden arribar a molta més gent afavorint l'aprenentatge. Punts a destacar: - Traduïda a 30 idiomes. - Applet's de física atòmica i nuclear.

Pàgina web	Anàlisis
http://phet.colorado.edu/en/simulations **Il·lustració 4:** Vista pàgina web PhET	Pàgina web que pertany a la Universitat de Colorado i que mitjançant a un projecte anomenat PhET han creat un repositori de simulacions molt extens que inclou experiments de matemàtiques, física, química etc... Les simulacions estan fetes en Flash[2] i Java, aquestes ultimes utilitzen el protocol JNLP (explicat en apartats posteriors) per a descarregar les aplicacions en lloc d'utilitzar applet's incrustats a la web oferint poc dinamisme. També cal destacar la gran varietat de nivells des de els experiments senzills i divertits als més complexes. Punts a destacar: - Utilització de Flash i Java amb JNLP - Calcificació per tipus i nivell dels experiments. - Gran varietat d'experiments.

[2] Adobe Flash es tracta essencialment d'una aplicació de creació i manipulació vectorial (relativament bàsic) i de maneig de codi en forma d'estudi d'animació que treballa sobre "fotogrames" i està destinat a la producció i lliurament de contingut interactiu per a les diferents audiències al voltant del món sense importar la plataforma. Adobe Flash utilitza gràfics vectorials i imatges de mapa de bits, so, codi de programa, flux de vídeo i àudio bidireccional, en sentit estricte. Flash és l'entorn i Flash Player és el programa de màquina virtual utilitzat per executar els arxius generats amb Flash, en paraules més senzilles, Adobe Flash crea i edita les animacions o arxius multimèdia i Adobe Flash Player les reprodueix.

Pàgina web	Anàlisis
www.um.es/fem/EjsWiki/ Il·lustració 5: Vista pàgina web EjS	Aquesta web no es un laboratori virtual, però es una pàgina que te molta importància en el mon dels laboratoris virtuals. Aquesta web explica que es el EjS (Easy Java Simulations) per part del seu creador Francisco Esquembre, es professor de la Universitat de Murcia. Aquesta eina es usada en moltes pàgines com a eina de generació de simulacions, ja que esta orientada a usuaris sense coneixements de programació i sobre tot al mon de la docència. Amb aquesta eina es pot generar una simulació en qualsevol àmbit de la ciència, ja que gràcies als elements visuals com els de modelatge es pot simular qualsevol experiència. A més a les simulacions creades es poden acompanyar d’una explicació del que s’està representant i com es comporta matemàticament.

Pàgina web	Paràmetres a avaluar sobre els laboratoris analitzats					Total	Nota final
	Gràfics	Ciència	Diversitat	Tecnologia	Visualització de dades		
http://www.myphysicslab.com/	9	8	6	9	9	41	8.2
http://www.phy.ntnu.edu.tw/ntnujava/	7	8	7	7	8	37	7.4
http://www.walter-fendt.de/ph14e/	5	6	8	6	7	32	6.4
http://phet.colorado.edu/en/simulations	8	7	9	9	7	40	8

Després de realitzar les qualificacions sobre els paràmetres seleccionats, la pàgina amb millor puntuació en tots els àmbits es: http://www.myphysicslab.com/. Els paràmetres que s'han utilitzat per avaluar els laboratoris son: els gràfics, que inclou la qualitat gràfica dels experiments, fluïdesa de les animacions, semblança amb la realitat etc... la ciència, en poques paraules quant de ciència hi ha en les simulacions, no només la quantitat sinó la qualitat d'aquesta; la diversitat: la varietat de tipologies dels experiments o simulacions de les pàgines analitzades, quants més tipus més bona puntuació se li ha donat; la tecnologia, part molt important de com estan fetes les simulacions, el que s'ha valorat positivament es la diversitat de tecnologies com la utilització d'últimes tecnologies; per últim, com es visualitzen les dades que es generen als experiments, es un punt molt important, tant o més que els gràfics dels experiments.

Així, per fer la conclusió de l'anàlisi la pàgina guanyadora: es simple, potent amb molta sortida d'informació, gràcies a la manera de representar les dades i amb una física complerta i de bon nivell. Es de totes la que té mes interacció amb l'usuari, ja que permet quasi manipular qualsevol experiment com si fos amb les mans i per tant es la més realista de totes.

2 Introducció a la física dels models implementats

2.1 Simulació física. Mètodes numèrics[ii]:

Amb els models presentats és clar que desitgem animar objectes. Per poder fer-ho necessitem un motor de càlcul numèric. Aquest serà el motor encarregat de calcular el nou valor per a totes les posicions , velocitats i altres propietats de cadascuna de les nostres partícules i/o objectes a simular.

Per tal d’implementar la dinàmica de partícules sota forces donades ens caldrà avaluar equacions. De fet són equacions diferencials de primer ordre.

2.1.1 Avaluació de les equacions diferencials

Una equació diferencial descriu la relació entre una funció qualsevol i el valor de les seves derivades. Per verificar l'equació hauran de complir-se algunes condicions. En la majoria dels casos parlarem del que es coneix com a "problemes de valor inicial" on el comportament del sistema pot resumir-se com una funció derivada del tipus:

$$x' = f(x, t)$$

Aquí f és una funció coneguda que depèn de x i de t, x és l'estat del sistema i x' la seva derivada. Això és una equació diferencial ordinària (EDO).

Los condicions inicials; indiquem:

$$x(t_0) = x_0$$

L'estat del sistema per a un cert temps. Després el sistema haurà d'anar variant el seu comportament segons les forces externes que se li apliquin.

Intentarem obtenir la derivada d'una funció en diferents intervals discrets ja que no podem fer-ho de forma contínua. Aplicarem per tant una solució numèrica per resoldre aquest tipus de problemes

Tenim una funció derivada f que ens avalua els canvis en l'estat del sistema x, que són el que denominem Δx, i ho fa al llarg d'un interval de temps anomenat Δt. Llavors el que farem serà incrementar l'estat, o sigui x, amb el valor calculat Δx, obtenint així el nou estat.

Els mètodes numèrics avaluen una o més derivades en un mateix interval de temps. Això els fa diferents quant a eficiència i robustesa. A continuació es mostren dos mètodes numèrics, el d'Euler i el de Runge-Kutta, el primer es l'escollit per implementar les simulacions d'aquest projecte. El segon es un altre mètode més precís però que te un cost computacional més elevat.

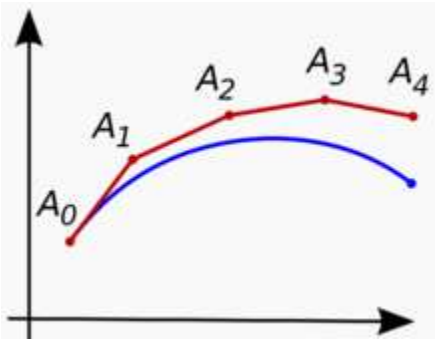
2.2 Mètode d'Euler[iii]

En matemàtiques i ciència computacional, el mètode d'Euler és un procés numèric per a resoldre equacions diferencials ordinàries (EDOs) amb un valor inicial donat. És el cas més bàsic de mètode explícit d'integració numèrica per a equacions diferencials ordinàries. El mètode pren el nom del seu autor, Leonhard Euler.

2.2.1 Descripció geomètrica informal

Considerem el problema de calcular l'àrea d'una corba desconeguda que comença en un punt donat i satisfà una equació diferencial donada. L'equació diferencial es pot interpretar com una fórmula que permet calcular el pendent de la recta tangent a qualsevol punt de la corba, a partir de la posició d'aquest punt.

La idea és que, encara que la corba és inicialment desconeguda, el seu punt inicial, que notarem per A_0, és conegut. Llavors, a partir de l'equació diferencial, es pot calcular el pendent de la corba a A_0, i per tant, la recta tangent en el punt inicial.



Il·lustració 6: Mètode d'Euler

Avancem un petit pas al llarg d'aquesta recta tangent cap a un punt A_1. Si suposem que A_1 encara es troba sobre la corba, es pot utilitzar el mateix raonament que pel punt A_0. Després d'alguns passos, es calcula la corba poligonal $A_0A_1A_2A_3A_4$. En general, aquesta corba no divergeix gaire de la corba original desconeguda, i l'error entre totes dues corbes es pot reduir si la mida del pas és prou petita i l' interval de computació és finit.

2.2.2 Derivació

Volem aproximar la solució del problema de valor inicial:

$$y'(t) = f\big(t, y(t)\big), \;\; y(t_0) = y_0$$

utilitzant els dos primers termes de la sèrie de Taylor de y, que representa l'aproximació lineal al voltant del punt $t_0, y(t_0)$. Un pas del mètode d'Euler des de t_n cap a $t_{n+1} = \; t_n + h$ és:

$$y_{n+1} = y_n + hf(t_n, y_n)$$

El mètode d'Euler és explícit, és a dir, la solució y_{n+1} és una funció explícita de $y_i$per $i \leq n$.

Encara que el mètode d'Euler integra una EDO de primer ordre, qualsevol EDO d'ordre N es pot representar com una EDO de primer ordre amb més d'una variables introduint $N - 1$ variables addicionals, $y', y'', \ldots, y^{(N)}$, i formulant N equacions de primer grau amb aquestes noves variables. El mètode d'Euler es pot aplicar al vector $\vec{y}(t) = (y(t), y'^{(t)}, y''^{(t)}, \ldots, y^{(N)}(t)$ per integrar el sistema d'ordre més elevat.

2.2.3 Error

La magnitud dels errors generats pel mètode d'Euler es pot demostrar per comparació amb una sèrie de Taylor de y. Si assumim que $f(t)$ i $y(t)$ es coneixen exactament al temps t_0, llavors el mètode d'Euler dóna la solució aproximada al temps $t_0 + \; h$ com:

$$y(t_0 + h) = y(t_0) + \; h\, f\big(t_0.y(t_0)\big) = \; y(t_0) + hy'(t_0)$$

(la segona igualtat prové que i satisfà l'equació diferencial $y' = f(t, y)$). En comparació, la sèrie de Taylor a h sobre $t_0$dóna:

$$y(t_0 + h) = y(t_0) + hy'(t_0) + \frac{1}{2}\, h^2 y''(t_0) + O(h^3)$$

L'error introduït pel mètode d'Euler ve donat per la diferència entre aquestes equacions:

$$\frac{1}{2}\, h^2 y''(t_0) + O(h^3)$$

Per *h* petita, l'error dominant per pas és proporcional a h^2. Per resoldre el problema sobre un rang donat de *t*, el nombre de passos necessaris és proporcional a 1 / *h*, per tant s'espera que l'error total al final del temps fixat sigui proporcional a *h* (error per pas multiplicat pel nombre de passos). Per aquesta raó, es diu que el mètode d'Euler és de primer ordre. Això fa que el mètode d'Euler sigui menys precís (per petites h) que altres tècniques d'ordres més alts com el mètode de Runge-Kutta.

2.3 Mètode Runge-Kutta[iv]

El mètode de Runge-Kutta és un mètode genèric de resolució numèrica d'equacions diferencials. Aquest conjunt de mètodes va ser inicialment desenvolupat al voltant de l'any 1900 pels matemàtics C. Runge i M. W. Kutta.

Runge-Kutta no és només un mètode sinó una important família de mètodes iteratius tant implícits com a explícits per aproximar les solucions d'equacions diferencials ordinàries (E.D.O´s).

Els mètodes de Runge-Kutta aconsegueixen l'exactitud del procediment d'una sèrie de Taylor sense requerir el càlcul de derivades superiors. Existeixen moltes variacions, però totes es poden denotar en la forma generalitzada de l'equació:

$$y_{i+1} = y_i + \varphi(x_i, y_i, h)h$$

On $\varphi(x_i, y_i, h)$ és coneguda com a funció increment, la qual pot interpretar-se com un pendent representatiu sobre l’ interval.

$$\phi = a_1 k_1 + a_2 k_2 + \cdots + a_n k_n$$

On les a són constants i les k són:

$k_1 = f(x_i, y_i)$

$k_2 = f(x_i + p_1 h\,, y_i + q_{11} k_1 h)$

$k_3 = f(x_i + p_2 h\,, y_i + q_{21} k_1 h + q_{22} k_2 h)$

Observem que les k són relacions de recurrència, això és, k_1 apareix en l'equació per a k_2, la qual apareix en l'equació per a k_3, etc.

Com que cada k i és una avaluació funcional, aquesta recurrència fa que els mètodes Runge-Kutta siguin eficients per a la programació. Existeixen diversos tipus de mètodes Runge-Kutta en emprar diferents nombres de termes en la funció increment com l'especificada per n, per n = 1, és el mètode de Euler. Una vegada es tria n, s'avaluen les a, p i q en igualar la funció increment als termes en la sèrie de Taylor

2.4 Moviment rectilini uniforme (MRU):

És aquell moviment rectilini que es produeix quan el cos es mou amb velocitat constant. Tindrà, com qualsevol altre moviment rectilini, l'acceleració normal nul·la; per altra banda, també serà nul·la l'acceleració tangencial, ja que la velocitat es manté constant.

En la vida quotidiana és molt difícil trobar objectes que es moguin amb velocitat constant, ja que sempre hi ha un fregament que fa que la velocitat disminueixi. Però és molt útil com a aproximació estudiar el moviment amb velocitat constant, donada la seva senzillesa que ens aporta informació bàsica per entendre moviments més complexos.

Com que la velocitat és una magnitud vectorial, direm que es manté constant quan no canvia:

-El mòdul, és a dir, el valor numèric de la seva magnitud.

-La direcció, que és la recta que segueix en l’espai.

-El sentit, que es determina per l’origen i final del vector.

Per tant, perquè no canviï la direcció, la trajectòria ha de ser una línia recta i el mòbil sempre ha d’anar en el mateix sentit i amb el mateix valor de la velocitat.

Es pot modelitzar el comportament d'un mòbil en MRU a partir d'una única funció o equació, que ens dóna la posició en cada instant en funció del temps:

$$x(t) = x_0 + v\,(t - t_0)$$

On:

x = Posició del mòbil respecte el sistema de referència triat

x_0 = Posició inicial (quan $t = t_0$)

v = velocitat, que es manté constant

t = temps

t_0= temps inicial (quan $x = x_0$)

2.5 Moviment rectilini uniformement accelerat (MRUA)

És aquell moviment rectilini on la velocitat varia de manera constant, o sigui, existeix una acceleració tangencial constant i diferent de 0. Com en el cas anterior l'acceleració normal serà nul·la.

Aquest moviment es pot modelitzar amb dues equacions bàsiques en cada direcció del pla. Limitarem l’estudi als casos en què l’acceleració és constant, és a dir, als moviments rectilinis uniformement accelerats.

2.5.1 Equació de la posició:

Aquesta funció o equació ens dóna la posició d'un mòbil en MRUA en cada instant, en funció del temps:

$$x(t) = x_0 + v_0\,(t - t_0) + \frac{1}{2}a\,(t - t_0)^2$$

On:

x = Posició del mòbil respecte el sistema de referència triat.

x_0 = Posició inicial (quan $t = t_0$).

v_0 = Velocitat inicial del mòbil (quan $t = t_0$).

a = Acceleració, que es manté constant.

t = Temps.

t_0 = Temps inicial (quan $x = x_0$ i $v = v_0$).

2.5.2 Equació de la velocitat:

En aquest cas, com que la trajectòria és rectilínia, no canvia ni la direcció ni el sentit, només el mòdul de la velocitat: per això només hi ha acceleració tangencial. Aquesta expressió relaciona la velocitat del mòbil amb el temps.

$$v(t) = v_0 + a\,(t - t_0)$$

On:

v = Velocitat del mòbil en funció del temps t.

v_0 = Velocitat inicial (quan t = t_0).

a = Acceleració, que es manté constant.

t = Temps.

t_0 = Temps inicial (quan $x = x_0$).

2.6 Moviment harmònic simple (MHS):

A cinemàtica, s'anomena moviment harmònic a aquell moviment on un mòbil passa periòdicament pels mateixos punts de la seva trajectòria.

Quan el període d'un moviment harmònic es constant, s'anomena moviment harmònic simple (MHS).

Un MHS es produeix típicament quan un mòbil al qual s'ha donat una certa amplitud i/o velocitat inicials es troba sotmès als efectes d'una força conservativa. L'exemple més típic és una molla que puja i baixa quan l'estirem una mica.

2.6.1 Magnituds i caracterització del MHS

- Amplitud (A): és l'amplada de l'oscil·lació, mesurada des del centre fins a un dels extrems.

-Freqüència angular (ω): també anomenada velocitat angular o pulsació, ens indica quina és la velocitat del moviment. Es mesura en 'radiants per segon' (rad / s).

-Període (T): és el temps emprat pel mòbil per completar un cicle del MHS.

-Freqüència (f): és el nombre de cicles per unitat de temps. És l'invers del període. Es mesura en 'Hertz' (Hz = s^{-1}).

$$\omega = 2\pi f = \frac{2\pi}{T}$$

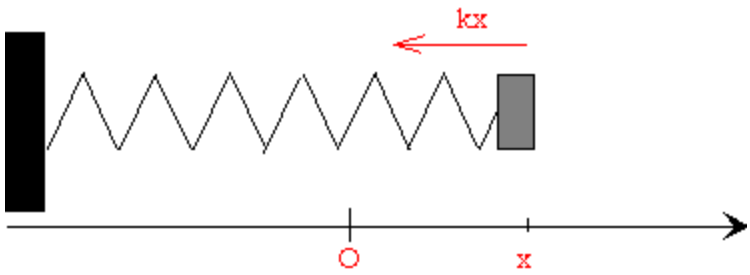
$$T = \frac{1}{f} = \frac{2\pi}{\omega} \Leftrightarrow f = \frac{1}{T} = \frac{\omega}{2\pi}$$

-Fase (φ): Ens indica en quin moment del cicle es troba el mòbil. De fet, és el terme que es troba dins de l'argument de la funció trigonomètrica en les equacions del model matemàtic.

2.6.2 Oscil·lador lliure:

Per explicar el comportament d'un oscil·lador lliure utilitzarem com a model una partícula de massa m unida a una molla[3] elàstica de constant k.

Quan una partícula es desplaça x de la posició d'equilibri, actua sobre ella una força que és proporcional al desplaçament x, i de sentit contrari a aquest, tal com es mostra en la següent figura:



Il·lustració 7: Oscil·lador lliure

Equació del moviment

L'equació del moviment s'escriu:

$$ma = -kx$$

Tenint en compte que l'acceleració és la derivada segona de la posició x, podem expressar l'equació del moviment com a equació diferencial de segon ordre.

[3] Es coneix com a molla a un operador elàstic capaç d'emmagatzemar energia i desprendre's d'ella sense sofrir deformació permanent quan cessen les forces o la tensió a les quals és sotmès. Són fabricats amb materials molt diversos, tals com a acer al carboni, acer inoxidable, acer al crom-silici, crom-vanadi, bronzes, plàstic, entre uns altres, que presenten propietats elàstiques i amb una gran diversitat de formes i dimensions.

$$\frac{d^2x}{dt^2} + \omega_0^2 x = 0 \qquad \omega_0^2 = \frac{k}{m}$$

ω_0 es denomina freqüència pròpia o natural de l'oscil·lador harmònic.

L'avantatge d'expressar les oscil·lacions en termes d'una equació diferencial és que podem establir analogies entre sistemes físics oscil·lants completament diferents: mecànics elèctrics, hidràulics, etc.

La solució d'aquesta equació diferencial és l'equació de M.H.S.

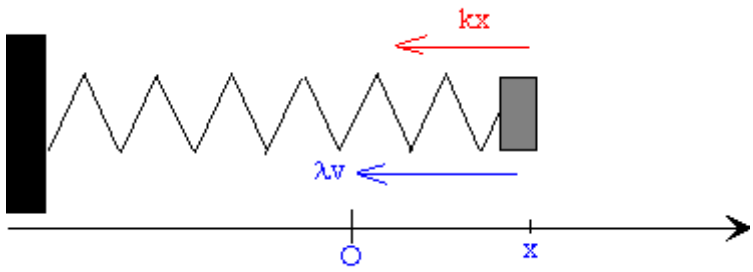
$$x = A\ \sin(\omega_0 t + \ \varphi)$$

$$v = \frac{dx}{dt} = A\ \omega_0 \cos(\omega_0 t + \ \varphi)$$

$$a = \frac{dv}{dt} = -A\ \omega_0^2 \sin(\omega_0 t + \ \varphi)$$

2.6.3 Oscil·lador esmorteït:

L'experiència ens mostra que l'amplitud d'un cos vibrant tal com una molla o un pèndol, decreix gradualment fins que es deté.



Il·lustració 8: Oscil·lador esmorteït

Per explicar l'esmorteïment, podem suposar que a més de la força elàstica $F = -kx$, actua una altra força oposada a la velocitat $Fr = -\lambda v$, on λ és una constant que depèn del sistema físic particular.

Tot cos que es mou dins d'un fluid viscós en règim laminar experimenta una força de fregament proporcional a la velocitat i de sentit contrari a aquesta[4] [v].

Equació del moviment

L'equació del moviment s'escriu:

$$ma = -kx - \lambda v$$

Expressem l'equació del moviment en forma d'equació diferencial, tenint en compte que l'acceleració és la derivada segona de la posició x, i la velocitat és la derivada primera de x.

$$\frac{d^2x}{dt^2} + 2\gamma\frac{dx}{dt} + \omega_0^2 x = 0 \quad \omega_0^2 = \frac{k}{m} \quad 2\gamma = \frac{\lambda}{m}$$

La solució de l'equació diferencial té la següent expressió:

$$x = Ae^{(-\gamma t)}\sin(\omega t + \varphi) \quad \omega^2 = \omega_0^2 - \gamma^2$$

$$v = \frac{dx}{dt} = -\gamma Ae^{(-\gamma t)}\sin(\omega t + \varphi) + Ae^{(-\gamma t)}\omega\cos(\omega t + \varphi)$$

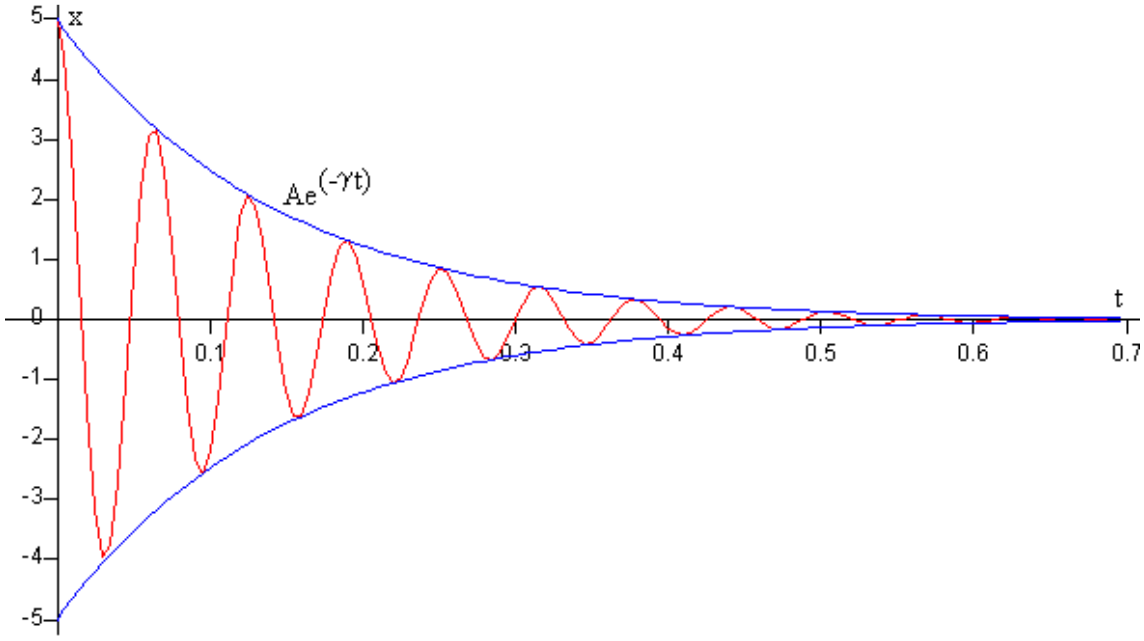
Les característiques essencials de les oscil·lacions esmorteïdes:

- L'amplitud de l'oscil·lació disminueix amb el temps.
- L'energia de l'oscil·lador també disminueix, a causa del treball de la força Fr de fregament viscós oposada a la velocitat.
- En l'espai de les fases $(v - x)$ el mòbil descriu una espiral que convergeix cap a l'origen.

Si l'esmorteïment és gran, g pot ser major que w_0, i w pot arribar a ser zero (oscil·lacions crítiques) o imaginari (oscil·lacions sobresmorteïdes). En tots dos casos, no hi ha oscil·lacions i la partícula s'aproxima gradualment a la posició d'equilibri. L'energia que perd la partícula que experimenta una oscil·lació esmorteïda és absorbida pel mitjà que l'envolta.

[4] La força de fregament és proporcional a la velocitat, i la seva expressió es denomina llei de Stokes:

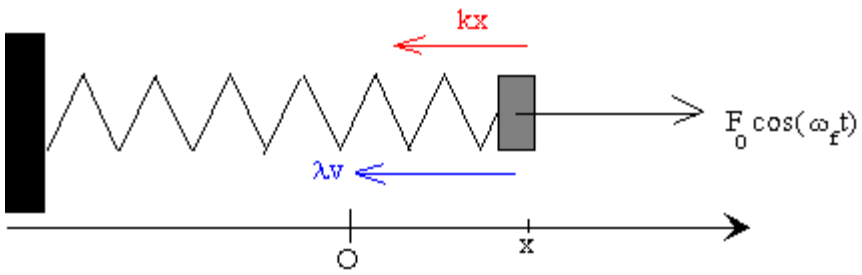
$$F_r = 6\pi R\eta v$$



Il·lustració 9: Equació amplitud oscil·lació esmorteïda.

2.6.4 Oscil·lador forçat:

Com hem vist als apartats anteriors, l'amplitud d'una oscil·lació esmorteïda decreix amb el temps. Al cap d'un cert temps (teòricament infinit), l'oscil·lador es deté en l'origen. Per mantenir l'oscil·lació és necessari aplicar una força oscil·lant.



Il·lustració 10: Oscil·lador forçat.

Les forces que actuen sobre la partícula són:

- La força que exerceix la molla $-kx$
- La força de fregament proporcional a la velocitat λv i de sentit contrari a aquesta.
- La força oscil·lant $F_0 cos(\omega_f t)$ de freqüència angular ω_f.

Equació del moviment

L'equació del moviment s'escriu:

$$ma = -kx - \lambda v + F_0 cos(\omega_f t)$$

Expressem l'equació del moviment en forma d'equació diferencial:

$$\frac{d^2x}{dt^2} + 2\gamma\frac{dx}{dt} + \omega_0^2 x = \frac{F_0}{m}\cos(\omega_f t) \quad \omega_0^2 = \frac{k}{m} \quad \gamma = \frac{\lambda}{2m}$$

- on ω_0 és la freqüència natural o pròpia de l'oscil·lador.
- ω_f és la freqüència angular de la força oscil·lant d'amplitud F
- γ és la constant d'esmorteïment, $\gamma < \omega_0$

La solució general de l'equació diferencial homogènia té la forma:

$$x_1 = (C\cos(\omega t) + D\sin(\omega t))e^{(-\gamma t)} \quad \omega = \sqrt{\omega_0^2 - \gamma^2}$$

On els coeficients C i D es determinen a partir de les condicions inicials. Una solució particular de l'equació diferencial completa té la forma:

$$x_2 = A\cos(\omega_f t) + B\sin(\omega_f t)$$

Obtindrem els valors d' A i B fent que compleixi l'equació diferencial lineal completa

$$A = \frac{F(\omega_0^2 - \omega_f)}{m\left(\left(\omega_0^2 - \omega_f^2\right)^2 + 4\gamma^2\omega_f^2\right)} \qquad B = \frac{2\gamma\omega_f F}{m\left(\left(\omega_0^2 - \omega_f^2\right)^2 + 4\gamma^2\omega_f^2\right)}$$

La solució general de l'equació diferencial completa és la suma de la solució general de l'homogènia més la solució particular $x = x_1 + x_2$.

$$x = (C\cos(\omega t) + D\sin(\omega t))e^{(-\gamma t)} + A\cos(\omega_f t) + B\sin(\omega_f t)$$

El primer terme, descriu l'estat transitori que desapareix al cap de cert temps (teòricament infinit) i depèn de les condicions inicials. El segon terme, descriu l'estat estacionari.

La velocitat és:

$$v = \frac{dx}{dt} = \begin{cases} -A\,\omega_f \sin(\omega_f t) + \omega_f\, B\cos(\omega_f t) \\ -\gamma(C\cos(\omega t) + D\sin(\omega t))e^{(-\gamma t)} + (C\,\omega\,\sin(\omega t) + D\omega\cos(\omega t))e^{(-\gamma t)} \end{cases}$$

Si les condicions inicials són $t = 0$, $x = x_0$, $v = v_0$.

$$C = x_0 - A$$

$$D = \frac{1}{\omega}(v_0 + \gamma C - \omega_f B)$$

Les condicions inicials més senzilles són $x = 0$, i $dx/dt = 0$ en l'instant $t = 0$. La partícula de massa m parteix de l'origen amb velocitat inicial nul·la.

$$C = -\frac{F(\omega_0^2 - \omega_f^2)}{m\left((\omega_0^2 - \omega_f^2)^2 + 4\gamma^2\omega_f^2\right)} \qquad D = \frac{F\gamma(\omega_f^2 + \omega_0^2)}{\omega m\left((\omega_0^2 - \omega_f^2)^2 + 4\gamma^2\omega_f^2\right)}$$

La posició de la partícula x que experimenta una oscil·lació forçada en funció del temps t és:

$$x = -\frac{F}{m\left((\omega_0^2 - \omega_f^2)^2 + 4\gamma^2\omega_f^2\right)}\left((\omega_0^2 - \omega_f^2)(\cos(\omega_f t) - e^{(-\gamma t)}\cos(\omega t)) + 2\gamma\omega_f\left(\sin(\omega_f t) - \frac{\omega_0^2 + \omega_f^2}{2\omega\omega_f}e^{(-\gamma t)}\sin(\omega t)\right)\right)$$

Casos particulars, sense fregament

$\boldsymbol{\gamma}$=0

Prop de la ressonància[5]

$\boldsymbol{\omega_f} = \boldsymbol{\omega_0}$+ε amb ε< $\boldsymbol{\omega_0}$

[5] La ressonància és un fenomen que es produeix quan un cos capaç de vibrar (o moure's) és sotmès a l'acció d'una força periòdica, el període de vibració coincideix amb el període d'oscil·lació característic d'aquest cos.
En aquestes circumstàncies el cos vibra (o es mou), augmentant de forma progressiva l'amplitud del moviment després de cadascuna de les actuacions successives de la força.

Aquest efecte pot ser destructiu en alguns materials rígids com el got que es trenca quan una soprano canta i arriba i sosté la freqüència de ressonància d'aquest. Per la mateixa raó, no es permet el pas per ponts de tropes marcant el pas, ja que poden entrar en ressonància i ensorrar-lo.

Prop de la ressonància ω_f=$\omega_0 + \varepsilon$ amb $\varepsilon < \omega_0$. Si no hi ha fregament, la freqüència angular de l'oscil·lació esmorteïda ω, és igual a la de l'oscil·lació lliure, ω_0. Com la diferència ε entre la freqüència ω_f de la força oscil·lant i la de ressonància ω_0 és petita, es pot fer la següent aproximació:

$$\omega_0^2 - \omega_f^2 = (\omega_0 + \omega_f)(\omega_0 - \omega_f) \approx -2\omega_0\varepsilon$$

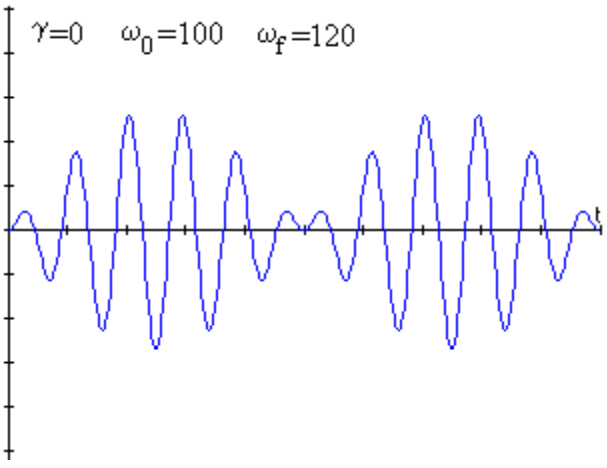
Emprant la següent relació trigonomètrica:

$$\cos A - \cos B = 2\sin\left(\frac{B+A}{2}\right)\sin\left(\frac{B-A}{2}\right)$$

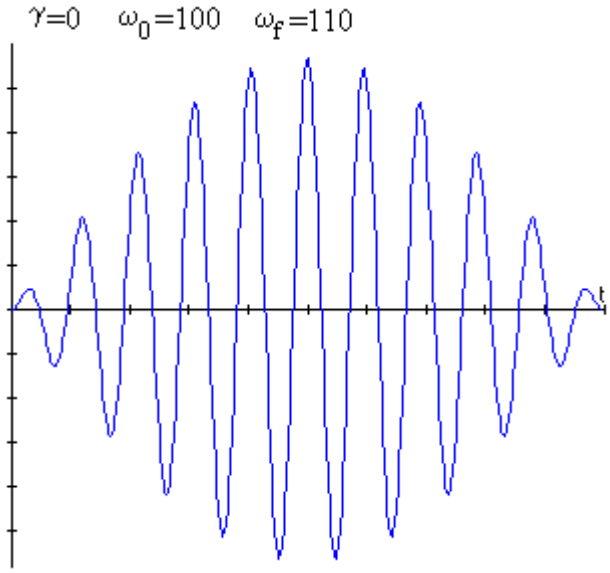
s'obté finalment:

$$x = \frac{F}{m\omega_0\epsilon}\sin\frac{\varepsilon t}{2}\sin(\omega_0 t)$$

L'amplitud (el producte dels dos primers termes) i la freqüència de les pulsacions es determinen pel grau de proximitat a la freqüència de ressonància tal com es pot apreciar en les figures.



Il·lustració 12: Amplitud en dos períodes de ressonància



Il·lustració 11: Amplitud màxima en un període

En ressonància

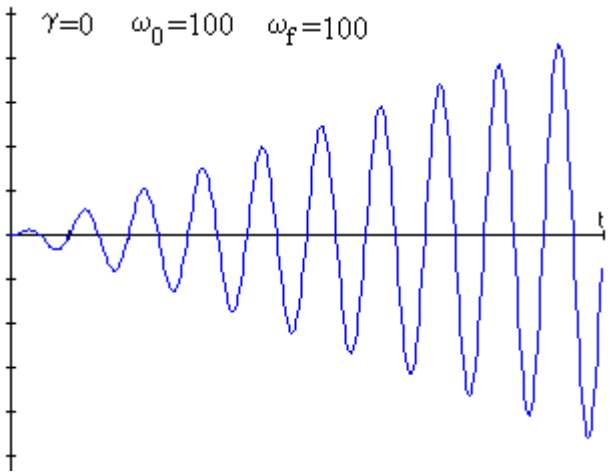
$$\boldsymbol{\omega_f = \omega_0}$$

En la fórmula del desplaçament $x(t)$ prenem el següent límit:

$$\lim_{\varepsilon \to 0} \sin \frac{\varepsilon t}{2} \longrightarrow \frac{\varepsilon t}{2}$$

L'amplitud creix linealment amb el temps, com pot veure's en la figura

$$x = \frac{F}{2m\omega_0} t \sin(\omega_0 t)$$



Il·lustració 13: Amplitud creixent per ressonància

Casos particulars, amb fregament

$$\boldsymbol{\gamma} \neq 0$$

Prop de la ressonància

$$\boldsymbol{\omega_f = \omega_0} + \varepsilon \text{ amb } \varepsilon < \boldsymbol{\omega_0}$$

Si la constant d'esmorteïment $\boldsymbol{\gamma}$ és petita, la freqüència angular de l'oscil·lació esmorteïda ω és gairebé igual a la de l'oscil·lació lliure ω_0, de manera que $\omega \approx \omega_0$. Com la diferència ε entre la freqüència ω_f de la força oscil·lant i la de ressonància ω_0 és petita, es pot fer la següent aproximació:

$$\omega_0^2 - \omega_f^2 = (\omega_0 + \omega_f)(\omega_0 - \omega_f) \approx -2\omega_0 \varepsilon$$

La solució de l'equació diferencial completa amb aquestes aproximacions ens queda:

$$x = \frac{F}{m(4\omega_0^2\varepsilon + 4\gamma^2\omega_0^2)}\Big(-2\omega_0\varepsilon\big(\cos(\omega_f t) - e^{(-\gamma t)}\cos(\omega_0 t)\big) + 2\gamma\omega_0\big(\sin(\omega_f t) - e^{(-\gamma t)}\sin(\omega_0 t)\big)\Big)$$

Canviant l'ordre dels termes:

$$x = \frac{F}{2m(\varepsilon^2 + \gamma^2)}\Big(-e^{(-\gamma t)}(-\varepsilon\cos(\omega_0 t) + \gamma\sin(\omega_0 t)) + \big(-\varepsilon\cos(\omega_f t) + \gamma\sin(\omega_0 t)\big)\Big)$$

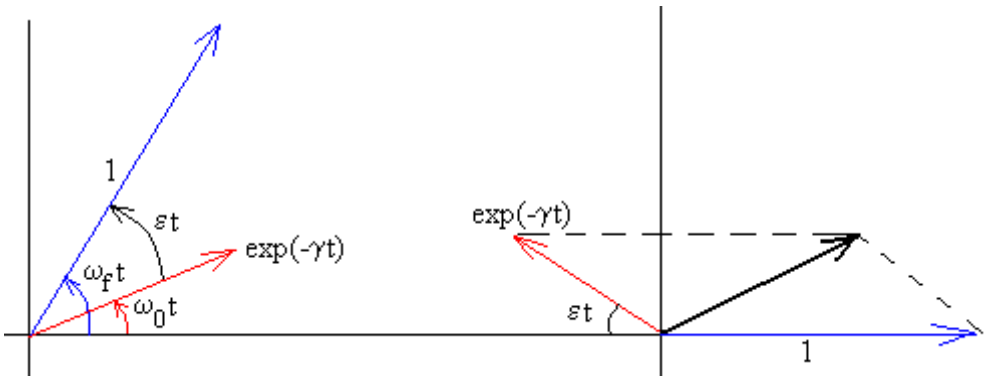
Podem escriure la suma de sinus i cosinus dels termes entre parèntesis de la següent manera:

$$-\varepsilon\cos(\omega_0 t) + \gamma\sin(\omega_0 t) = \sqrt{\varepsilon^2 + \gamma^2}\sin(\omega_0 t + \varphi) \qquad \tan\varphi = \frac{-\varepsilon}{\gamma}$$

I de manera similar, el segon terme entre parèntesis de freqüència angular ω_f

$$x = \frac{F}{2m\omega_0\sqrt{\varepsilon^2 + \gamma^2}}\big(-e^{(-\gamma t)}\sin(\omega_0 t + \varphi) + \sin\big(\omega_f t + \varphi\big)\big)$$

Tenim la composició de dues MHS de freqüències quasi iguals ω_0 i ω_f, una d'amplitud variable $e^{(-\gamma t)}$ i un altra d'amplitud constant 1.



Il·lustració 14: Diagrama vectorial del MHS

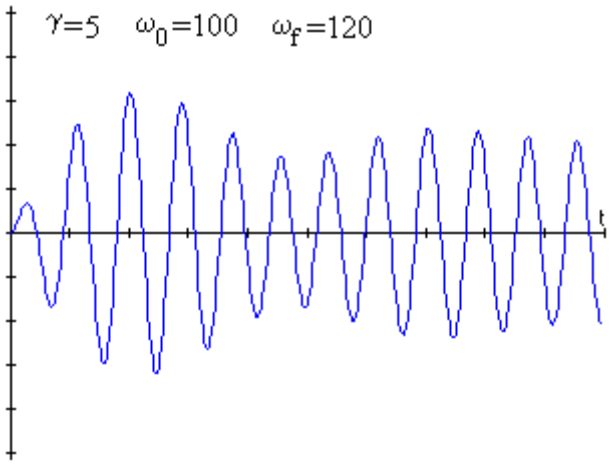
Mitjançant diagrames vectorials podem obtenir el MHS resultant el qual te una freqüència angulat aproximadament ω_0 i amb una amplitud que es aproximadament la diagonal de la dreta.

$$x = \frac{F}{2m\omega_0\sqrt{\varepsilon^2 + \gamma^2}}\Big(\sqrt{1 - 2e^{(-\gamma t)}\cos(\varepsilon t) + e^{(-2\gamma t)}}\sin\big(\omega_0 t + \varphi(t)\big)\Big)$$

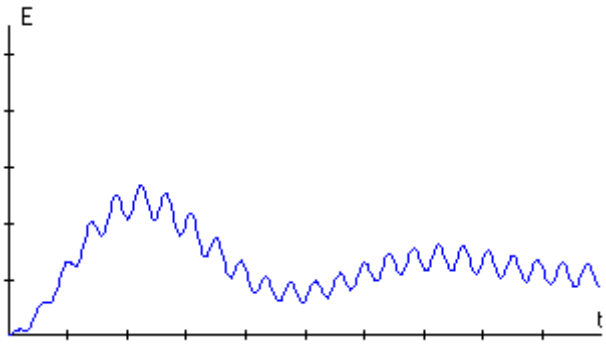
La fase de l'oscil·lació $\varphi(t)$ varia lentament amb el temps. L'amplitud, l'arrel quadrada que multiplica a la funció si, varia lentament amb la diferència de freqüències $\varepsilon = \omega_f - \omega_0$, al voltant del valor mitjà

$$\frac{F}{2m\omega_0\sqrt{\varepsilon^2+\gamma^2}}$$

L'amplitud de l'oscil·lació s'apropa gradualment a aquest valor mitjà a mesura que transcorre el temps tal com es pot veure en la figura. Durant l'estat transitori, l'amplitud pot aconseguir valors de fins a gairebé dues vegades l'amplitud de les oscil·lacions estables.



Il·lustració 15 Amplitud del oscil·lador fins arribar a l'estabilitat.



Il·lustració 16 Energia del oscil·lador en funció del temps

En la última figura , s'observa l'energia total I de l'oscil·lador (suma de cinètica i potencial) en funció del temps. La representació gràfica ens suggereix que l'energia de l'oscil·lador es descriu millor en termes de valors mitjans durant el període d'una oscil·lació. El valor mitjà de l'energia tendeix cap a un valor constant quan t es fa gran.

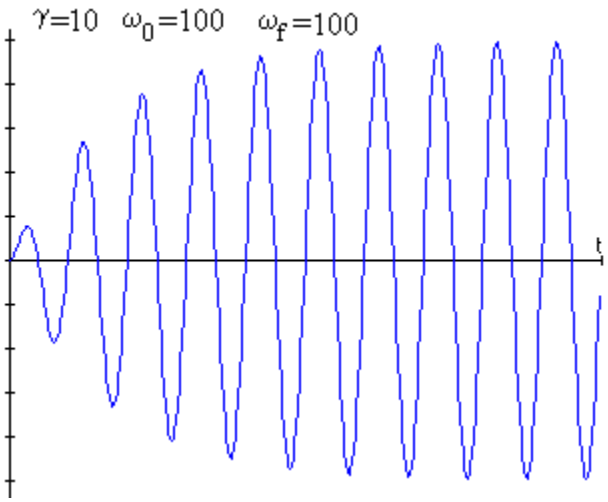
En ressonància

$$\boldsymbol{\omega_f = \omega_0}$$

En la formula del desplaçament $x(t)$ es simplifica notablement

$$x = \frac{F}{2my\omega_0}\left(1 - e^{(-\gamma t)}\right)\sin(\omega_0 t)$$

L'amplitud creix lentament, fins que s'apropa al valor asimptòtic $\frac{F}{2my\omega_0}$ determinat pel coeficient de fregament, tal com pot veure's en la figura.

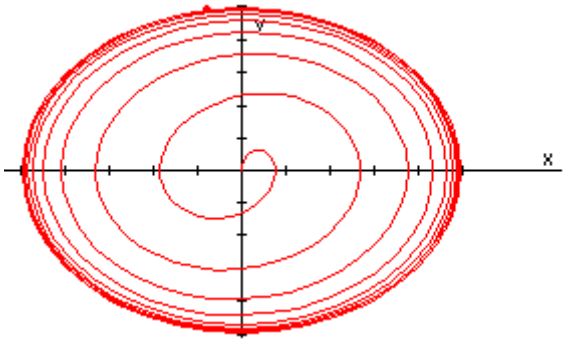


Il·lustració 17 Amplitud creixent de l'oscil·lador,

En la següent figura, observem la representació del la trajectòria de la partícula en l'espai de les fases[6], l'espai x-v (posició-velocitat). La partícula surt de l'origen i descriu una espiral que tendeix cap a una el·lipse límit, que s'aconsegueix en l'estat estacionari.

[6] En física i matemàtiques l'espai de fases és un espai matemàtic on es representen tots els possibles estats d'un sistema, de manera que cada un d'aquests possibles estats correspongui a un únic punt d'aquest espai. Cada eix d'aquest espai correspon a una variable o grau de llibertat del sistema. Cal no confondre el terme amb «diagrama de fases», que s'acostuma a reservar per a les representacions de les diverses fases d'un sistema termodinàmic.

L'espai de fases queda definit pel conjunt de variables que descriuen els graus de llibertat. Cada punt de l'espai correspon a un estat del sistema, i la seva evolució, en forma de trajectòria, dóna una imatge gràfica del que està passant. La dimensió de l'espai de fases coincideix amb el nombre de graus de llibertat del sistema i, per tant, amb la seva dimensionalitat dinàmica.



Il·lustració 18 Espai de fases x-v (posició - velocitat).

2.7 Camp elèctric

En física, el camp elèctric és el camp generat per un objecte carregat elèctricament, aquest camp genera una força que actua sobre d'altres objectes també carregats elèctricament. El concepte de camp elèctric va ser introduït per Michael Faraday[vi].

El camp elèctric és un vector que en unitats del SI, s'expressa en Newton[vii] per Coulomb[viii] ($N\ C^{-1}$) o, de manera equivalent, en Volt[ix] per metre ($V\ m^{-1}$). Els camps elèctrics es creen al voltant de qualsevol càrrega elèctrica. El sentit i direcció de les línies del camp en un punt donat de l'espai es defineixen com el sentit de la força que s'exerciria sobre una càrrega positiva de prova situada en aquell punt. Els camps elèctrics contenen energia elèctrica amb una densitat d'energia que és proporcional al quadrat de la intensitat del camp.

En el cas que l'objecte carregat elèctricament que produeix el camp elèctric tingui velocitat constant, el camp elèctric va acompanyat d'un camp magnètic. En el cas que la velocitat no sigui constant, sinó que presenti un moviment accelerat, es parla de camp electromagnètic. En general els camps elèctrics i magnètics no són fenòmens separats; en funció del punt de referència, allà on un observador aprecia un camp elèctric, un altre observarà un camp magnètic i un tercer una barreja de tots dos. En mecànica quàntica, les alteracions dels camps electromagnètics són anomenades fotons, i la seva energia és un quàntum.

2.7.1 Definició electrostàtica

El camp elèctric es defineix com la força elèctrica per unitat de càrrega. La direcció i el sentit del camp es pren de la direcció i el sentit que la força exerciria sobre una càrrega de prova positiva. Segons això el camp elèctric generat per una càrrega positiva és radial i cap enfora mentre que per una de negativa és radial i cap endins.

El camp elèctric es defineix com la constant de proporcionalitat entre la càrrega i la força, altrament dit, la força per unitat de càrrega:

$$\vec{E} = \frac{\vec{F}}{q}$$

On:

F és la força elèctrica donada per la Llei de Coulomb[x],

q és la càrrega d'una "càrrega de prova".

2.7.2 Moviment en un camp elèctric

Una partícula carregada que està en una regió on hi ha un camp elèctric, experimenta una força igual al producte de la seva càrrega per la intensitat del campo elèctric:

$$\vec{F}e = q\,\vec{E}$$

- Si la càrrega es positiva, experimenta una força en el sentit del camp.

- Si la càrrega es negativa, experimenta una força en sentit contrari al camp.

Si el camp és uniforme, la força és constant i també ho és, l'acceleració. Aplicant les equacions del moviment rectilini uniformement accelerat, obtenim la velocitat de la partícula en qualsevol instant o després d'haver-se desplaçat una determinada distància:

$$\vec{a} = \frac{q\,\vec{E}}{m} \qquad v = v_0 + at \qquad x = v_0 t + \frac{1}{2}at^2$$

2.8 Camp magnètic

En física clàssica, el camp magnètic és un camp vectorial, un vector a cada punt de l'espai per a qualsevol moment, que en unitats del Sistema Internacional es mesura en Tesles (un Tesla[xi] és un Newton-segon per Coulomb-metre $\frac{N \cdot s}{C \cdot m}$). Aquest camp vectorial té la propietat d'ésser solenoïdal.

2.8.1 Camp solenoïdal

Un camp que les seves fonts escalars son nul·les en tots els punts de l'espai:

$$\nabla \cdot \vec{B} = 0 \ \forall r$$

Un camp solenoïdal es caracteritza perquè les seves línies de camp no poden convergir ni divergir de cap punt, no poden tenir extrems localitzats. Això fa que les línies únicament puguin:

- ser tancades, o
- anar de l'infinit a l'infinit, o
- donar voltes sobre si mateixes, sense arribar a tancar-se, com en una madeixa.

El camp magnètic pot ser definit i mesurat per mitjà d'un petit dipol magnètic, com per exemple un petit imant. El camp magnètic exerceix un moment de força sobre els dipols magnètics que tendeix a adquirir la mateixa direcció que el camp, de manera idèntica al cas d'una brúixola, i a més la magnitud d'aquest parell de forces és proporcional a la magnitud del camp magnètic. En conseqüència, per tal de mesurar el camp magnètic a un punt determinat de l'espai s'hi pot posar un petit imant que pugui girar lliurement, com una brúixola; la direcció ver la que apunti serà la del camp magnètic i la ràtio de la màxima magnitud del parell de forces del moment magnètic de l' imant serà la magnitud del camp.

Hi ha altres vies diferents, físicament equivalents, per tal de definir el camp magnètic, com per exemple a través de la força de Lorentz[xii] o a través d'una solució de les equacions de Maxwell[xiii].

De qualsevol de les definicions que es pugui utilitzar es deriva que el camp vectorial magnètic, essent un producte vectorial, és un pseudovector o vector axial.

2.8.2 Vector axial

Un vector axial o pseudovector és una magnitud física que presenta propietats de covariància o transformació sota reflexions anòmala, presentant violacions aparents de la paritat física.

L'explicació d'aquest estrany comportament és que realment no qualsevol 3-tupla de components forma un vector físic. En particular, qualsevol magnitud física definida mitjançant el producte vectorial de dos vectors físics genuïns és un vector axial o pseudovector. Les components d'un vector axial $\vec{a} = (a_1, a_2, a_3)$ tridimensional admeten ser expressades com:

$$a_i = \sum_{j,k} \varepsilon_{ijk} b_j c_k$$

On:

ε_{ijk} es el símbol de Levi-Civita[7], totalment antisimètric.

$\vec{b} = (b_1, b_2, b_3)$ es un vector ordinari no axial.

$\vec{c} = (c_1, c_2, c_3)$ es un vector ordinari no axial.

En mecànica relativista els vectors axials són tractats com la part espacial d'un tensor antisimètric. Més concretament un vector axial resulta ser la part espacial del dual de Hodge[8] del corresponent tensor antisimètric.

[7] En matemàtiques, i en particular en càlcul tensorial, es defineix el símbol de Levi-Civita, també cridat el símbol de permutació, nomenat així per Tullio Levi-Civita. S'utilitza en moltes àrees de les matemàtiques i en física. Per exemple, en àlgebra lineal, el producte vectorial de dos vectors es pot escriure com:

$$\vec{a} \times \vec{b} = \begin{vmatrix} e_1 & e_2 & e_3 \\ a_1 & a_2 & a_3 \\ b_1 & b_2 & b_3 \end{vmatrix} = \sum_{i=1}^{3} \left(\sum_{j,k=1}^{3} \varepsilon_{ijk} a_j b_k \right) e_i$$

[8] En matemàtiques, l'operador estrella de Hodge en l'espai vectorial V és un operador lineal en l'àlgebra exterior de V, intercanviant els subespais de k-vectors i el de n−k-vectors on n = dim V, per 0 ≤ k ≤ n. Un exemple comú de l'operador estrella és l'espai euclidià tridimensional dotat de la mètrica ordinària. De

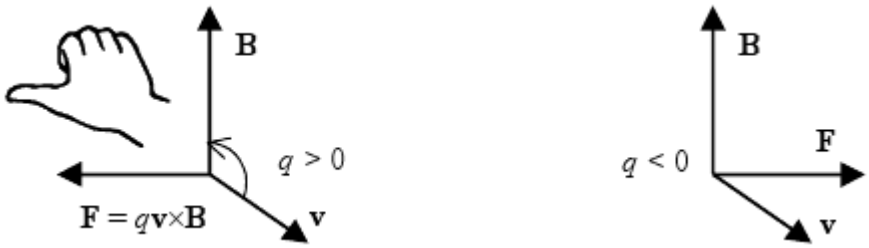
2.8.3 Moviment en un camp magnètic

Una partícula que es mou en un camp magnètic experimenta una força:

$$\vec{F}m = q\ (\vec{v}x\ \vec{B})$$

El resultat d'un producte vectorial és un vector de:

- mòdul igual al producte dels mòduls pel sinus de l'angle comprès $\left(\widehat{\mathrm{q}\vec{\mathrm{v}},\vec{\mathrm{B}}}\right)$
- direcció perpendicular al plànol format pels vectors velocitat $\vec{v}$ i camp $\vec{B}$.
- I el sentit s'obté per la denominada regla del llevataps .Si la càrrega és positiva el sentit és el del producte vectorial vB, com en la figura esquerra. Si la càrrega és negativa el sentit de la força és contrari al del producte vectorial vB:



Il·lustració 19 Regla de la mà dreta.

Una partícula carregada descriu òrbita circular en un camp magnètic uniforme. El radi d'aquesta òrbita, s'obté a partir de l'equació de la dinàmica del moviment circular uniforme: força igual a massa per acceleració normal.

$$Fm = m\ \frac{v^2}{r} \qquad qvB = \ m\ \frac{v^2}{r} \qquad r = \ \frac{m\ v}{qB}$$

fet el producte vectorial no és una altra cosa que el dual de Hodge del producte exterior de dues formes diferencials construïdes a partir dels vectors, formalment el producte vectorial resulta ser:

$$\vec{a}\ \times\ \vec{b} = *\left(\phi_{\vec{a}} \wedge \phi_{\vec{b}}\right)$$

2.8.4 Moviment circular uniforme

En física, el moviment circular uniforme descriu el moviment d'un cos movent-se, amb velocitat constant, una trajectòria circular.

Encara que la celeritat de l'objecte és constant, la seva velocitat no ho és: La velocitat, una magnitud vectorial, tangent a la trajectòria, en cada instant canvia de direcció. Aquesta circumstància implica l'existència d'una acceleració que, si bé en aquest cas no varia al mòdul de la velocitat, sí varia la seva direcció.

2.8.5 Angle i velocitat angular

L'angle abastat en un moviment circular és igual al quocient entre la longitud de l'arc de circumferència recorreguda i el radi.

La longitud de l'arc i el radi de la circumferència són magnituds de longitud, per la qual cosa el desplaçament angular és una magnitud adimensional[9], anomenada radiant. Un radiant és un arc de circumferència de longitud igual al radi de la circumferència, i la circumferència completa té 2π radiants.

La velocitat angular és la variació del desplaçament angular per unitat de temps:

$$\omega = \frac{d\varphi}{dt}$$

Partint d'aquests conceptes s'estudien les condicions del moviment circular uniforme, quant a la seva trajectòria i espai recorregut, velocitat i acceleració, segons el model físic cinemàtic.

[9] En física, química, enginyeria i altres ciències aplicades es denomina magnitud adimensional a tota aquella magnitud que manca d'una magnitud física associada. Així, serien magnituds adimensionals totes aquelles que no tenen unitats, o les unitats de les quals poden expressar-se com a relacions matemàtiques pures.

2.8.6 Vector de la posició

Es considera un sistema de referència en el plànol xy, amb vectors unitaris en el sentit d'aquests eixos $(O; i, j)$. La posició de la partícula en funció de l'angle de gir φ i del radi r és en un sistema de referència cartesià xy:

$$\begin{cases} x = r\ \cos\varphi \\ y = r\ \sin\varphi \end{cases}$$

En ser un moviment uniforme, a iguals increments de temps li corresponen iguals desplaçaments angulars, la qual cosa es tradueix en:

$$\omega = \frac{d\varphi}{dt} = \frac{\varphi}{t} \qquad \varphi = \omega t$$

De manera que el vector de posició de la partícula en funció del temps és:

$$\vec{r} = r\ cos\,\omega t\ \mathbf{i} + r\ \sin\omega t\ \mathbf{j}$$

sent:

r: és el vector de posició de la partícula.

r: és el radi de la trajectòria.

ω: és la velocitat angular (constant).

t : és el temps.

2.8.7 Vector de velocitat

La velocitat s'obté a partir del vector de posició mitjançant derivació:

$$\vec{v} = \frac{d\vec{r}}{dt} = -r\omega\ \sin(\omega t)\ \mathbf{i} + r\omega\ cos(\omega t)\,\mathbf{j}$$

El vector velocitat és tangent a la trajectòria, la qual cosa pot comprovar-se fàcilment efectuant el producte escalar r · v i comprovant que és nul.

2.8.8 Vector d’Acceleració

L'acceleració s'obté a partir del vector velocitat mitjançant derivació:

$$\vec{\mathrm{a}} = \frac{d\vec{\mathrm{v}}}{dt} = -r\omega^2 \cos(\omega t)\ \mathbf{i} - r\omega^2 \sin(\omega t)\,\mathbf{j}$$

de manera que:

$$\vec{\mathrm{a}} = -\omega^2\vec{\mathrm{r}}$$

Així doncs, el vector acceleració té la mateixa direcció i sentit oposat que el vector de posició, normal a la trajectòria i apuntant sempre cap al centre de la trajectòria circular. pel que acostumem a referir-nos a ella com a acceleració normal o centrípeta.

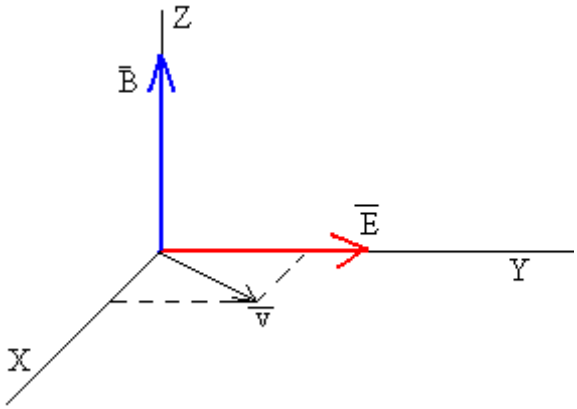
El mòdul de l'acceleració és el quadrat de la velocitat angular pel radi de gir, encara que ho podem expressar també en funció de la celeritat v, de la partícula, ja que, en virtut de la relació $\mathrm{v} = \omega\mathrm{r}$, resulta:

$$\vec{a} = \omega^2 r = \frac{\vec{v}^2}{\vec{r}}$$

Aquesta acceleració és l'única que experimenta la partícula quan es mou a velocitat constant en una trajectòria circular, per la qual cosa la partícula haurà de ser atreta cap a en centre mitjançant una força centrípeta que l'a part d'una trajectòria rectilínia, com correspondria per la llei d'inèrcia.

2.9 Moviment en camps elèctric i magnètic creuats

Suposem que el camp magnètic B té l'adreça de l'eix Z, el camp elèctric I l'adreça de l'eix Y, i el vector velocitat v està en el plànol XY. La partícula carregada part de la posició inicial (x_0, y_0) amb velocitat inicial (vx_0, vy_0).



Il·lustració 20 Definició vectorial de forces

La força que exerceix el camp elèctric E sobre una carrega q és:

$$\vec{F}e = q\,\vec{E}$$

La força que exerceix el camp magnètic B sobre una partícula de càrrega q la velocitat de la qual és v és:

$$\vec{F}m = q\,(\vec{v}\,x\,\vec{B})$$

L'equació del moviment és:

$$m\frac{d^2\boldsymbol{r}}{dt^2} = q\vec{E} + q\vec{v}\,\times\,\vec{B}$$

Les components d'E, B i v són:

$B\ (0,0,B)$

$E\ (0,E,0)$

$v\ (v_{0x},v_{0y},0)$

$$m\left(\frac{dv_x}{dt}\boldsymbol{i} + \frac{dv_y}{dt}\boldsymbol{j} + \frac{dv_z}{dt}\boldsymbol{k}\right) = qE\boldsymbol{j} + q\begin{vmatrix}\boldsymbol{i} & \boldsymbol{j} & \boldsymbol{k}\\ v_x & v_y & 0\\ 0 & 0 & B\end{vmatrix}$$

S'obté el següent sistema d'equacions diferencials:

$$\frac{dv_x}{dt} = \frac{qB}{m}\, vy$$

$$\frac{dv_y}{dt} = \frac{qE}{m} - \frac{qB}{m}\, vx$$

$$\frac{dv_z}{dt} = 0$$

La velocitat al llarg de l'eix Z és constant i igual a la velocitat inicial, $\mathrm{v_z} = \mathrm{v_{0z}} = 0$. Es denomina freqüència de gir w al quocient $\mathrm{w} = \mathrm{qB/m}$, que és la velocitat angular d'un partícula carregada en un camp magnètic uniforme. Aïllem $\mathrm{v_y}$ en la primera equació i la introduïm en la segona. Obtenim l'equació diferencial de segon ordre.

$$\frac{d^2\, v_x}{dt^2} + w^2 v_x = \frac{q^2 EB}{m^2}$$

La solució d’aquesta equació diferencial es de la forma:

$$v_x = C\cos(wt) + D\sin(wt) + c$$

Introduint $\mathrm{v_x}$ en la equació diferencial determinant la solució particular c de la equació diferencial de segon ordre:

$$v_x = C\cos(wt) + D\sin(wt) + \frac{E}{B}$$

Calculem la component $\mathrm{v_y}$ de la velocitat de la partícula

$$v_x = \frac{1}{\omega}\frac{dv_x}{dt} = -C\sin(wt) + D\cos(wt)$$

Les constants C i D es determinen a partir de les condicions inicials. En l'instant t=0, les components de la velocitat de la partícula són ($\mathrm{v_{0x}}$, $\mathrm{v_{0y}}$).

$$v_x = \left(v_{0x} - \frac{E}{B}\right)\cos(wt) + vo_y\sin(wt) + \frac{E}{B}$$

$$v_y = -\left(v_{0x} - \frac{E}{B}\right)\sin(wt) + vo_y\cos(wt)$$

Per simplificar i generalitzar les expressions de les components de la velocitat, denominats velocitat de deriva: $v_d = \frac{E}{B}$ .Les expressions de v_x i v_y quedaran com segueix:

$$v_x = (v_{0x} - v_d)\, cos\,(\omega t) + v_{0y}\, sen(\omega t) + v_d$$
$$v_y = -(v_{0x} - v_d) \cdot sen(\omega t) + v_{0y}\, cos(\omega t)$$

Sabent que en l'instant $t = 0$, la posició de la partícula és (x_0, y_0), calculem la coordenada x del centre de la bola integrant l'expressió de la velocitat v_x en funció del temps, el mateix per a l'ordenada y.

$$x = x_0 + \frac{1}{\omega}\left\{(v_{0x} - v_d)\, sen(\omega t) + v_{0y}\,(1 - cos(\omega t))\right\} + v_d t$$

$$y = y_0 + \frac{1}{\omega}\left\{(v_{0x} - v_d)(\, cos(\omega t) - 1) + v_{0y}\, sin(\omega t)\right\}$$

Podem escriure de forma alternativa aquesta equació de la forma:

$$x - x_0 - \frac{v_{0y}}{\omega} - v_d t = \frac{1}{\omega}\left\{(v_{0x} - v_d)\, sen(\omega t) + v_{0y}\, cos(\omega t)\right\}$$

$$y - y_0 + \frac{1}{\omega}(v_{0x} - v_d) = \frac{1}{\omega}\left\{(v_{0x} - v_d)\, cos(\omega t) + v_{0y}\, sin(\omega t)\right\}$$

Elevant al quadrat i sumant:

$$\left(x - x_0 - \frac{v_{0y}}{\omega} - v_d t\right)^2 + \left(y - y_0 + \frac{v_{0x}}{\omega} - \frac{v_d}{\omega}\right)^2 = \frac{1}{\omega}\left\{(v_{0x} - v_d)^2 + {v_{0y}}^2\right\}$$

Es tracta de l'equació d'una circumferència centrada en el punt (a, b) i de radi R_c:

$$(x - a)^2 + (y - b)^2 = {R_c}^2$$

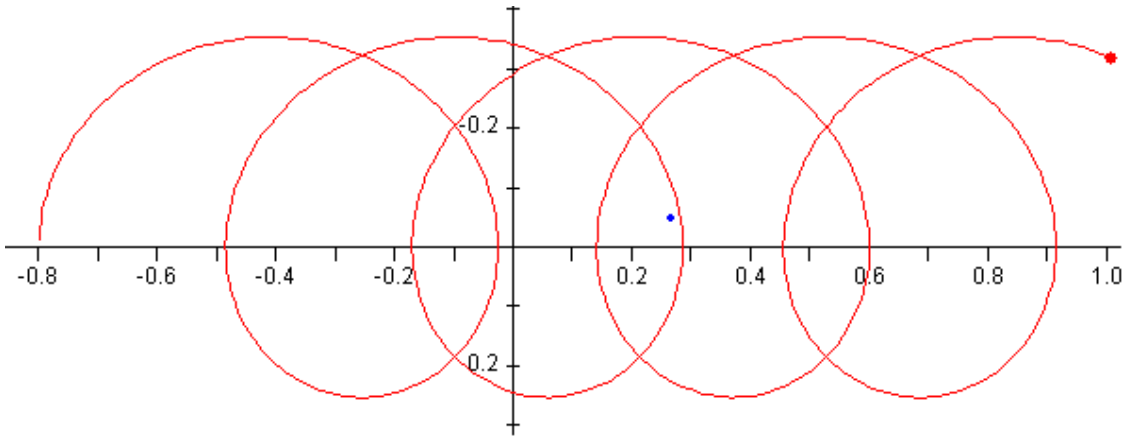
On:

$$a = x_0 + \frac{v_{oy}}{\omega} + v_d t$$

$$\mathrm{b} = \mathrm{y}_0 - \frac{\mathrm{v_{ox}}}{\omega} + \frac{\mathrm{v_d}}{\omega}$$

$$R_c = \frac{1}{\omega}\sqrt{(v_{0x} - v_d)^2 + {v_{0y}}^2}$$

El centre de la circumferència es mou al llarg de l'eix X (no al llarg del pla inclinat) amb velocitat $\mathrm{v_d}$:

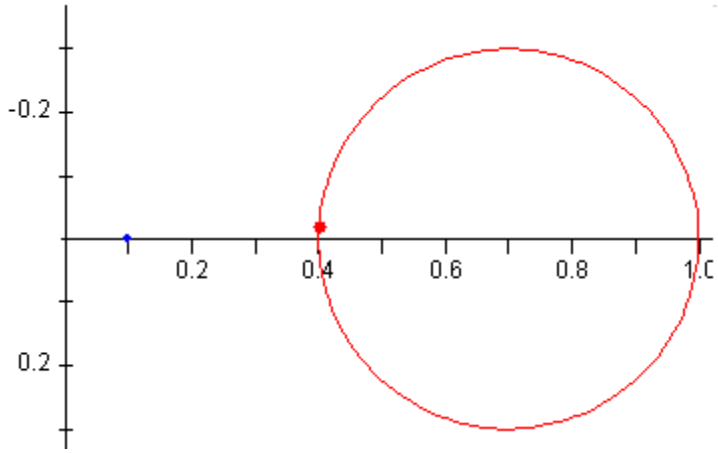


Il·lustració 21 Moviment d'una partícula amb camp elèctric i magnètic.

2.9.1 Casos particulars

1 - Quan el camp elèctric és nul $E = 0$, $v_d = 0$, la partícula descriu una circumferència, el centre del qual i radi són:

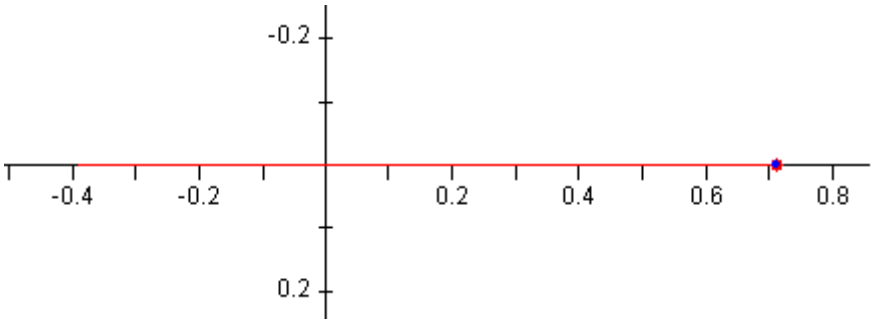
$$a = x_0 + \frac{v_{oy}}{\omega} \qquad \mathrm{b} = \mathrm{y}_0 - \frac{\mathrm{v_{ox}}}{\omega} \qquad R_c = \frac{v_{0x+}v_{oy}}{\omega}$$



Il·lustració 22 Moviment d'una partícula amb camp magnètic i camp elèctric nul.

2 - Si $v_{oy} = 0$, i $v_{ox} = v_d = {}^{E}/_{B}$, llavors la partícula es mou al llarg de l'eix X amb velocitat constant igual al quocient entre la intensitat del camp elèctric E i la intensitat del camp magnètic B. Aquest és el fonament d'un espectròmetre de masses[10].

$$x = x_0 + v_d t \qquad y = y_o$$



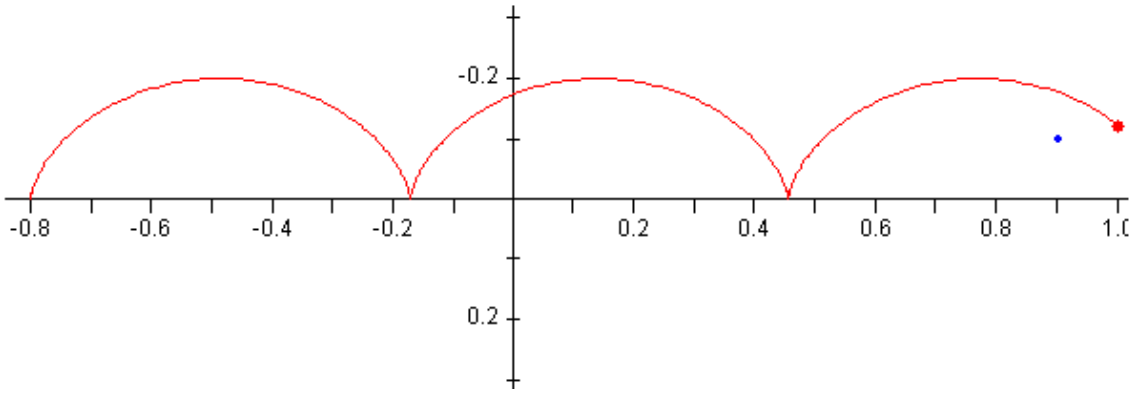
Il·lustració 23 Moviment sobre camp elèctric i magnètic però sense V_{0y}

3 – Quan la partícula pateix del repòs, $v_{ox} = v_{oy} = 0$ des de l'origen $x_o = y_o = 0$

[10] L' espectròmetre de masses és un instrument que permet analitzar amb gran precisió la composició de diferents elements químics i isòtops atòmics, separant els nuclis atòmics en funció de la seva relació massa-càrrega (m/z). Es pot utilitzar per identificar els diferents elements químics que formen un compost, o per determinar el contingut isotòpic de diferents elements en un mateix compost. Sovint es troba com a detector d'un cromatògraf de gasos, en una tècnica híbrida coneguda per les seves inicials en anglès, GC-MS.

$$x = \frac{\mathrm{v_d}}{\omega}(\omega t - \sin(\omega t)) \qquad y = \frac{\mathrm{v_d}}{\omega}(1 - \cos(\omega t))$$

Aquestes són les equacions paramètriques d'una cicloide generada per un punt de la vora d'un disc de ràdio $R = v_d/\omega = E/(\omega B)$, que roda sense lliscar, girant al voltant del seu eix amb velocitat angular ω i el centre de la qual es mou amb velocitat constant $v = R\omega = E/B$.



Il·lustració 24 Cicloide creat al no tenir velocitat inicial

2.10 Xocs de partícules

Un xoc físic o mecànic és una sobtada acceleració o desacceleració causada, per exemple, per impacte, per una gota d'aigua, per una explosió. Qualsevol tipus de contacte directe és en realitat un xoc, però el que ho caracteritza és la durada del contacte que generalment és molt curta i és quan es transmet la major quantitat d'energia entre els cossos.

2.10.1 Xocs elàstics

En física, es denomina xoc elàstic a una col·lisió entre dues o més cossos en la qual aquests no sofreixen deformacions permanents durant l'impacte. En una col·lisió elàstica es conserven tant el moment lineal com l'energia cinètica del sistema, i no hi ha intercanvi de massa entre els cossos, que se separen després del xoc.

2.10.2 Xocs inelàstics

Un xoc inelàstic és un tipus de xoc en el qual l'energia cinètica no es conserva. Com a conseqüència, els cossos que col·lisionen poden sofrir deformacions i augment de la seva temperatura. En el cas ideal d'un xoc perfectament inelàstic entre objectes macroscòpics, aquests romanen units entre si després de la col·lisió. El marc de referència del centre de masses permet presentar una definició més precisa.

La principal característica d'aquest tipus de xoc és que existeix una dissipació d'energia, ja que tant el treball realitzat durant la deformació dels cossos com l'augment de la seva energia interna s'obté a costa de l'energia cinètica dels mateixos abans del xoc. En qualsevol cas, encara que no es conservi l'energia cinètica, sí es conserva el moment lineal total del sistema.

Les col·lisions en les quals la energia no es conserva produeixen deformacions permanents dels cossos i es denominen inelàstiques.

2.10.3 Coeficient de restitució

El coeficient de restitució és una mesura del grau de conservació de l'energia cinètica en un xoc entre partícules clàssiques. En una col·lisió frontal alineada de dues esferes sòlides (com les que experimenten les boles de billar) les velocitats després del xoc estan relacionades amb les velocitats abans del xoc, per l'expressió:

$$e = -\frac{v_2 - v_1}{u_2 - u_1}$$

On:

u_1, u_2 són les velocitats de les partícules abans del xoc i v_1, v_2 després del xoc.

On e és precisament el coeficient de restitució, que pren valors entre 0 i 1. El valor 1 es dóna en un xoc perfectament elàstic, on es conserva tant el moment lineal com l'energia cinètica del sistema. El valor 0 es dóna en un xoc perfectament inelàstic, on només es conserva el moment lineal, una porció de l'energia cinètica inicial de les partícules es "consumeix" durant el xoc, convertint-se en energia de deformació plàstica i en so.

Il·lustració 25 Estats abans i després d'un xoc entre dos partícules.

Les velocitats de les esferes $v1 = -v$ i $v2 = v$ després del xoc estan relacionades amb les velocitats $u_1 = u$ i $u2 = -u$ abans del xoc mitjançant:

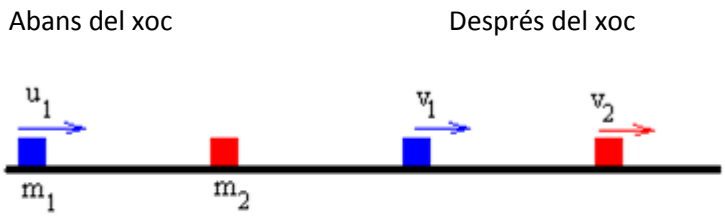
$$v1 - v2 = -e(u1 - u2)$$
$$-2v = -e(2u)$$

La velocitat v de les esferes després del xoc disminueix ja que el coeficient de restitució $e < 1$.

$$v = e\,u$$

2.10.4 Xocs frontals de partícules

En aquest apartat veurem un xoc frontal i elàstic entre dues partícules. En aquest cas, la primera partícula porta una velocitat u_1 i la segona està inicialment en repòs $u_2 = 0$.



Il·lustració 26 Comportament masses en un xoc de partícules,

1 Principi de conservació del moviment lineal[xiv]

$$m_1 u_1 = m_1 v_1 + m_2 v_2$$

2 En un xoc elàstic, l'energia cinètica inicial es igual a la final.

$$\frac{1}{2} m_1 u_1^2 = \frac{1}{2} m_1 v_1^2 + \frac{1}{2} m_2 v_2^2$$

Resolent aquest sistema de dues equacions amb dues incògnites, obtenim les velocitats v_1 i v_2 després del xoc.

$$v_1 = \frac{(m_1 - m_2) u_1}{m_1 + m_2}$$

$$v_2 = \frac{2 m_1 u_1}{m_1 + m_2}$$

Per a un sistema de dues partícules, la màxima velocitat que aconsegueix la segona partícula és $2u_1$ quan la massa de la segona partícula m_2 és molt petita comparada amb la massa de la primera partícula m_1. Ho podem apreciar millor si escrivim v_2 en funció de $x = m_2/m_1$.

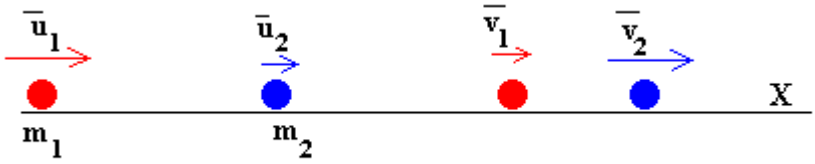
$$v_2 = \frac{2u_1}{1 + x}$$

$$\lim_{x \to 0} v_2 = 2u_1$$

Quan $m_1 = m_2$ la velocitat de la primera partícula després del xoc és zero $v_1 = 0$, la primera partícula es queda aturada i la segona partícula adquireix la velocitat (i l'energia) de la primera partícula, $v_2 = u_1$.

Aquest seria la representació del joc de billar quan les dues partícules tenen la mateixa massa i una d'elles esta en repòs.

Per generalitzar com es comporta aquest tipus de xoc introduirem que la segona partícula no esta en repòs:



Il·lustració 27 Comportament velocitats en un xoc de partícules.

1 Principi de conservació del moviment lineal

$$m_1u_1 + m_2u_2 = m_1v_1 + m_2v_2$$

2 En un xoc elàstic, l'energia cinètica inicial es igual a la final.

$$\frac{1}{2}m_1u_1^2 + \frac{1}{2}m_2u_2^2 = \frac{1}{2}m_1v_1^2 + \frac{1}{2}m_2v_2^2$$

Donats u_1 i u_2, les velocitats de les partícules m1 i m2 abans del xoc, podem calcular les velocitats de les partícules v_1 i v_2 després del xoc resolent el sistema de dues equacions amb dues incògnites.

Transformem les dues equacions, en les equivalents:

$$m_1(u_1 - v_1) = -m_2(u_2 - v_2)$$
$$m_1(u_1^2 - v_1^2) = -m_2(u_2^2 - v_2^2)$$

La diferència dels quadrats de dos nombres és igual al producte de la seva suma per la seva diferència:

$$m_1(u_1 - v_1) = -m_2(u_2 - v_2)$$
$$m_1(u_1 - v_1)(u_1 + v_1) = -m_2(u_2 - v_2)(u_2 + v_2)$$

Ens queda un sistema de dues equacions més fàcil de resoldre:

$$m_1(u_1 - v_1) = -m_2(u_2 - v_2)$$
$$(u_1 + v_1) = (u_2 + v_2)$$

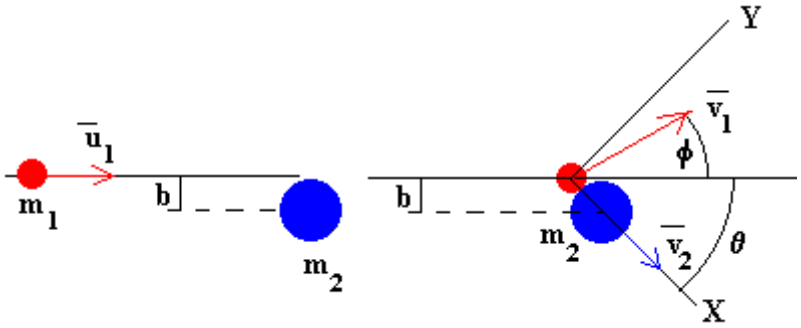
Aïllem les velocitats de les partícules després del xoc v_1 i v_2

$$v_1 = \frac{2m_2u_2 + (m_1 - m_2)u_1}{m_1 + m_2}$$

$$v_2 = \frac{2m_2u_2 + (m_1 - m_2)u_1}{m_1 + m_2}$$

2.10.5 Xocs bidimensionals

Suposem que xoquen dos discos o esferes de masses m_1 i m_2 i radis r_1 i r_2 .



Il·lustració 28 Comportament xoc bidimensional de partícules.

Es denomina paràmetre d'impacte b a la distància entre la direcció de la velocitat del primer disc u_1 i el centre del segon disc que suposem inicialment en repòs.

$$b = (r_1 + r_2)\sin\theta$$

Les velocitats dels discos abans del xoc respecte del sistema d'eixos X e Y:

$$\overrightarrow{u_1} = u_1 \cos\theta\, i + u_1 \cos\theta\, j$$

$$\overrightarrow{u_1} = 0$$

Les velocitats de discos després del xoc respecte del sistema d'eixos X e Y:

$$\overrightarrow{v_1} = v_1 \cos(\theta + \varphi)\, i + v_1 \sin(\theta + \varphi)\, j$$

$$\overrightarrow{v_2} = v_2\, i$$

El principi de conservació del moment lineal s'escriu:

$$m_1u_1\cos\theta = m_2v_2 + m_1v_1\cos(\theta + \varphi)$$
$$m_1u_1\sin\theta = m_1v_1\sin(\theta + \varphi)$$

El coeficient de restitució ens mesura el quocient canviat de signe, entre la velocitat relativa d'allunyament al llarg de l'eix X i la velocitat relativa d'aproximació al llarg del mateix eix.

$$e = \frac{v_2 - v_1 \cos(\theta + \varphi)}{u_1 \cos\theta}$$

Donat el paràmetre d'impacte b obtenim l'angle θ . De la segona i tercera equació, podem aïllar l'angle entre les direccions de les velocitats dels discos després del xoc:

$$\tan(\theta + \varphi) = \frac{m_1 + m_2}{m_1 - e m_2}\tan\theta \quad ; \quad v_1 = \frac{u_1 \sin\theta}{\sin(\theta + \varphi)} \quad ; \quad v_2 = \frac{m_1 u_1 (1 + e)\cos\theta}{m_1 + m_2}$$

Cas particular

Quan els discos tenen la mateixa massa $m_1 = m_2$, i el xoc és elàstic $e = 1$. L'angle que formen les direccions de les velocitats després del xoc és $\theta + \varphi = 90°$, i els seus mòduls són, respectivament:

$$v_1 = u_1 \sin\theta \quad v_2 = u_1 \cos\theta$$

$$b = (r_1 + r_2)\sin\theta$$

$$\varphi = 90° - \theta$$

3 Especificació

Una vegada tenim l'anàlisi de requeriments complet i detallat, el pas següent a seguir és l'especificació del sistema. En l'especificació l'objectiu és aclarir què és el que ha de fer el sistema, però sense entrar en els detalls tecnològics encara. És una de les parts difícils per a moltes persones, ja que requereix un esforç d'abstracció per fer l'especificació totalment independent de la tecnologia.

Per a aquest apartat usarem algunes de les eines típiques de les etapes d'especificació amb so el model de casos d'ús i el model conceptual de dades.

Llenguatge de modelatge: UML

Com a llenguatge de modelatge s'ha triat UML (Unified Modeling Language), es tracta del llenguatge de modelatge de sistemes de programació més conegut i usat en l'actualitat. És un llenguatge gràfic per visualitzar, especificar, construir i documentar un sistema. UML ofereix un estàndard per descriure un plànol del sistema (un model), que inclou aspectes conceptuals com els processos de negoci i funcions del sistema, i aspectes concrets com a expressions de llenguatges de programació, esquemes de bases de dades i components reutilitzables.

La utilització de UML proporciona una ajuda en la comunicació dels models del projecte, en proporcionar una semàntica suficient per capturar les decisions estratègiques i tàctiques, i proporcionar una notació robusta des de l'anàlisi al disseny del projecte.

Des de l'any 2005, UML és un estàndard aprovat per la ISO com ISO/IEC 19501:2005 Information technology — Open Distributed Processing — Unified Modeling Language (UML) Version 1.4.2 però la notació UML va ser proposada per Rational Programari Corporation i posteriorment estandarditzada pel Object Management Group (OMG). UML és el resultat de l'esforç d'unificació de les metodologies OMT, Booch i OOSE.

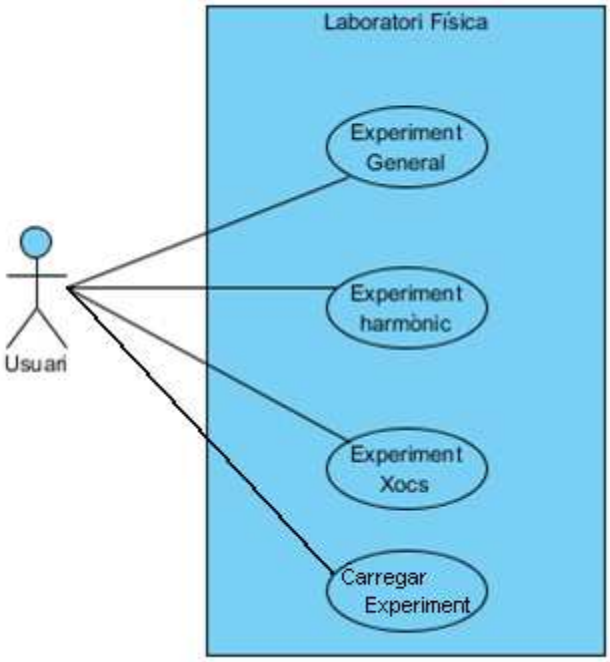
En l'actualitat, UML està en la seva versió d'implementació 2.0, en la qual existeixen 13 tipus diferents de diagrames estàndard dividits entre una visió estàtica i una altra dinàmica.

3.1 Especificació casos d'ús

Un cas d'ús és el conjunt d'escenaris relacionats que descriuen de quina manera els actors usen el sistema per aconseguir un determinat objectiu. Els casos d'ús es poden definir de diferents maneres, amb diverses dimensions. Les variants de les dimensions dels casos d'ús són l'estil, l'àmbit i el format.

Per a aquest apartat s'han definit els casos d'ús en l'àmbit del sistema programari, amb un estil essencial, on s'incorpora només la informació mínima necessària.

3.1.1 Diagrames de casos d’us



Il·lustració 29 Cas d’ús – Menú principal

Cas d'ús: Laboratori Física – Menú Principal
Actors: Usuari
Descripció:

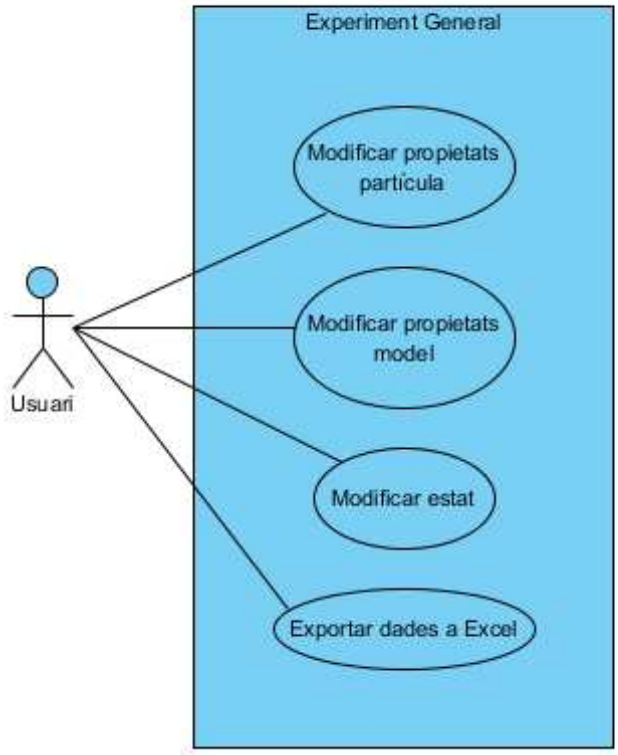
Escenari principal, curs típic de successos:

Usuari	Sistema
1- L’usuari accedeix a l’aplicació	2- El sistema una vegada carregat mostra un missatge on indica a l’usuari les tres opcions disponibles: experiment general, experiment harmònic, experiment xocs.
3- L’usuari escull quina de les tres opcions.	4- Una vegada l’usuari ha escollit el sistema canvia de pantalla i es mostra les opcions de tipus d’experiment escollit

Cas d'ús: Laboratori Física – Carregar Experiment
Actors: Usuari
Descripció:

Escenari principal, curs típic de successos:

Usuari	Sistema
1- L’usuari fa clic en “Obrir Experiment”	2- El sistema mostra una finestra de explorador de fitxers per que l’usuari indiqui la ruta.
3- L’usuari escull l’arxiu prèviament desat.	4- Una vegada l’usuari ha escollit el fitxer, el sistema reconeix a quin model pertany i l’obra recuperant les propietats que tenia en el moment de desar-ho.



Il·lustració 30 Cas d'ús –Experiment General

Cas d'ús: Experiment General – Modificar propietats partícula
Actors: Usuari
Descripció: Pantalla de l'experiment general

Escenari principal, curs típic de successos:

Usuari	Sistema
1- L'usuari modifica qualsevol de les propietats de la partícula (ja sigui massa, velocitats inicials, posicions etc...)	2- El sistema valida els valors introduïts en cada element a modificar, el sistema valida que es un valor correcte, tan en format com en magnitud.
3- L'usuari fa clic a "Assignar".	4- El sistema llegeix les dades introduïdes i les introdueix al model i actualitza la visualització

Cas d'ús: Experiment General – Modificar propietats model
Actors: Usuari
Descripció: Pantalla de l’experiment general

Escenari principal, curs típic de successos:

Usuari	Sistema
1- L’usuari modifica qualsevol de les propietats del model (camp Electric, gravetat, coeficient de restitució, etc...)	2- El sistema valida els valors introduïts en cada element a modificar, el sistema valida que es un valor correcte, tan en format com en magnitud.
3- L’usuari fa clic a “Assignar”.	4- El sistema llegeix les dades introduïdes i les introdueix al model i actualitza la visualització

Cas d'ús: Experiment General – Modificar estat
Actors: Usuari
Descripció: Pantalla de l’experiment general

Escenari principal, curs típic de successos:

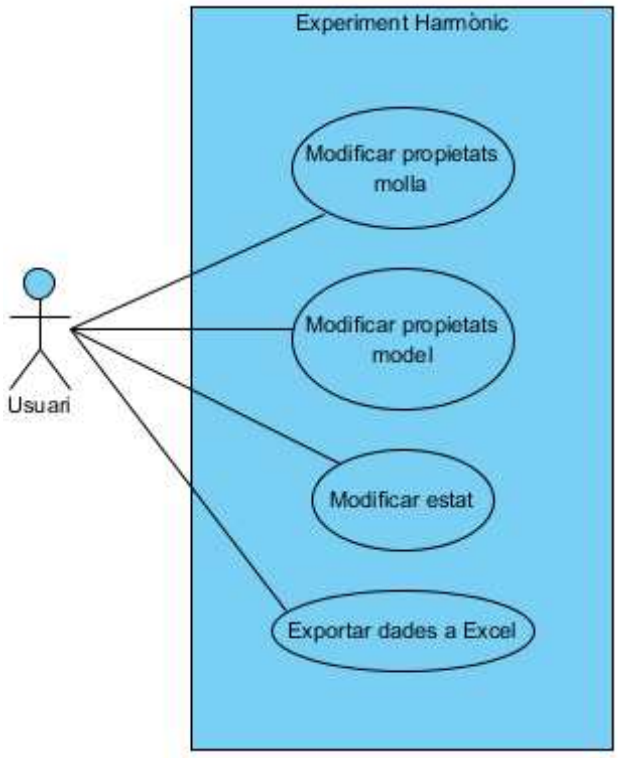
Usuari	Sistema
1- L’usuari fa clic sobre el botó “Aturat”.	2- El sistema activa el model i la vista, el model passa a estar “Activat”.
3- L’usuari fa clic sobre el botó “Activat”.	4- El sistema atura el model i la vista, el model passa a estar “Aturat”
5- L’usuari fa clic sobre el botó “Reinicia”	6- El sistema posa a l’estat inicial tant el model com la vista.

Cas d'ús: Experiment General – Exportar dades a Excel
Actors: Usuari
Descripció: Pantalla de l’experiment general

Condicions prèvies: L’experiment ha d’estar activat.

Escenari principal curs típic de successos:

Usuari	Sistema
1- L’usuari fa clic en “Exportar dades a Excel”	2- El sistema mostra una finestra amb l’opció de el nom i la ruta del fitxer a generar.
3- L’usuari introdueix la ruta i el nom del fitxer.	5- El sistema genera a la ruta indicada amb el nom que ha introduït l’usuari.



Il·lustració 31 Cas d’ús –Experiment Harmònic

Cas d'ús: Experiment Harmònic – Modificar propietats molla
Actors: Usuari
Descripció: Pantalla de l’experiment harmònic

Escenari principal, curs típic de successos:

Usuari	Sistema
2- L’usuari modifica qualsevol de les propietats de la molla (ja sigui elongació, k, etc...)	3- El sistema valida els valors introduïts en cada element a modificar, el sistema valida que es un valor correcte, tan en format com en magnitud.
4- L’usuari fa clic a “Assignar”.	5- El sistema llegeix les dades introduïdes i les introdueix al model i actualitza la visualització

Cas d'ús: Experiment Harmònic – Modificar propietats model
Actors: Usuari
Descripció: Pantalla de l’experiment general

Escenari principal, curs típic de successos:

Usuari	Sistema
2- L’usuari modifica qualsevol de les propietats del model (força externa, coeficient de restitució, etc...)	3- El sistema valida els valors introduïts en cada element a modificar, el sistema valida que es un valor correcte, tan en format com en magnitud.
4- L’usuari fa clic a “Assignar”.	5- El sistema llegeix les dades introduïdes i les introdueix al model i actualitza la visualització

Cas d'ús: Experiment Harmònic – Modificar estat
Actors: Usuari
Descripció: Pantalla de l'experiment general

Escenari principal, curs típic de successos:

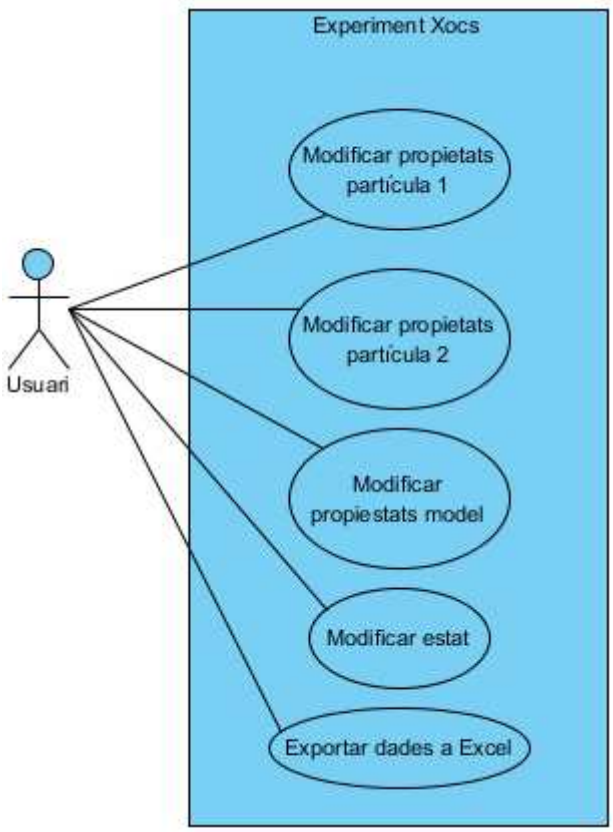
Usuari	Sistema
2- L'usuari fa clic sobre el botó "Aturat".	3- El sistema activa el model i la vista, el model passa a estar "Activat".
4- L'usuari fa clic sobre el botó "Activat".	6- El sistema atura el model i la vista, el model passa a estar "Aturat"
6- L'usuari fa clic sobre el botó "Reinicia"	7- El sistema posa a l'estat inicial tant el model com la vista.

Cas d'ús: Experiment General – Exportar dades a Excel
Actors: Usuari
Descripció: Pantalla de l'experiment general

Condicions prèvies: L'experiment ha d'estar activat.

Escenari principal curs típic de successos:

Usuari	Sistema
3- L'usuari fa clic en "Exportar dades a Excel"	4- El sistema mostra una finestra amb l'opció de el nom i la ruta del fitxer a generar.
5- L'usuari introdueix la ruta i el nom del fitxer.	7- El sistema genera a la ruta indicada amb el nom que ha introduït l'usuari.



Il·lustració 32 Cas d’ús –Experiment Xocs

Cas d'ús: Experiment Xocs – Modificar propietats partícula 1
Actors: Usuari
Descripció: Pantalla de l’experiment xocs

Escenari principal, curs típic de successos:

Usuari	Sistema
3- L’usuari modifica qualsevol de les propietats de la partícula 1 (ja sigui massa, velocitats inicials, posicions etc...)	4- El sistema valida els valors introduïts en cada element a modificar, el sistema valida que es un valor correcte, tan en format com en magnitud.
5- L’usuari fa clic a “Assignar”.	6- El sistema llegeix les dades introduïdes i les introdueix al model i actualitza la visualització

Cas d'ús: Experiment Xocs – Modificar propietats partícula 2
Actors: Usuari
Descripció: Pantalla de l’experiment xocs

Escenari principal, curs típic de successos:

Usuari	Sistema
4- L’usuari modifica qualsevol de les propietats de la partícula 1 (ja sigui massa, velocitats inicials, posicions etc...)	5- El sistema valida els valors introduïts en cada element a modificar, el sistema valida que es un valor correcte, tan en format com en magnitud.
6- L’usuari fa clic a “Assignar”.	7- El sistema llegeix les dades introduïdes i les introdueix al model i actualitza la visualització

Cas d'ús: Experiment Xocs – Modificar propietats model
Actors: Usuari
Descripció: Pantalla de l’experiment general

Escenari principal, curs típic de successos:

Usuari	Sistema
3- L’usuari modifica qualsevol de les propietats del model (camp Electric, gravetat, coeficient de restitució, etc...)	4- El sistema valida els valors introduïts en cada element a modificar, el sistema valida que es un valor correcte, tan en format com en magnitud.
5- L’usuari fa clic a “Assignar”.	6- El sistema llegeix les dades introduïdes i les introdueix al model i actualitza la visualització

Cas d'ús: Experiment Harmònic – Modificar estat
Actors: Usuari
Descripció: Pantalla de l'experiment general

Escenari principal, curs típic de successos:

Usuari	Sistema
4- L'usuari fa clic sobre el botó "Aturat".	5- El sistema activa el model i la vista, el model passa a estar "Activat".
6- L'usuari fa clic sobre el botó "Activat".	8- El sistema atura el model i la vista, el model passa a estar "Aturat"
7- L'usuari fa clic sobre el botó "Reinicia"	8- El sistema posa a l'estat inicial tant el model com la vista.

Cas d'ús: Experiment General – Exportar dades a Excel
Actors: Usuari
Descripció: Pantalla de l'experiment general

Condicions prèvies: L'experiment ha d'estar activat.

Escenari principal curs típic de successos:

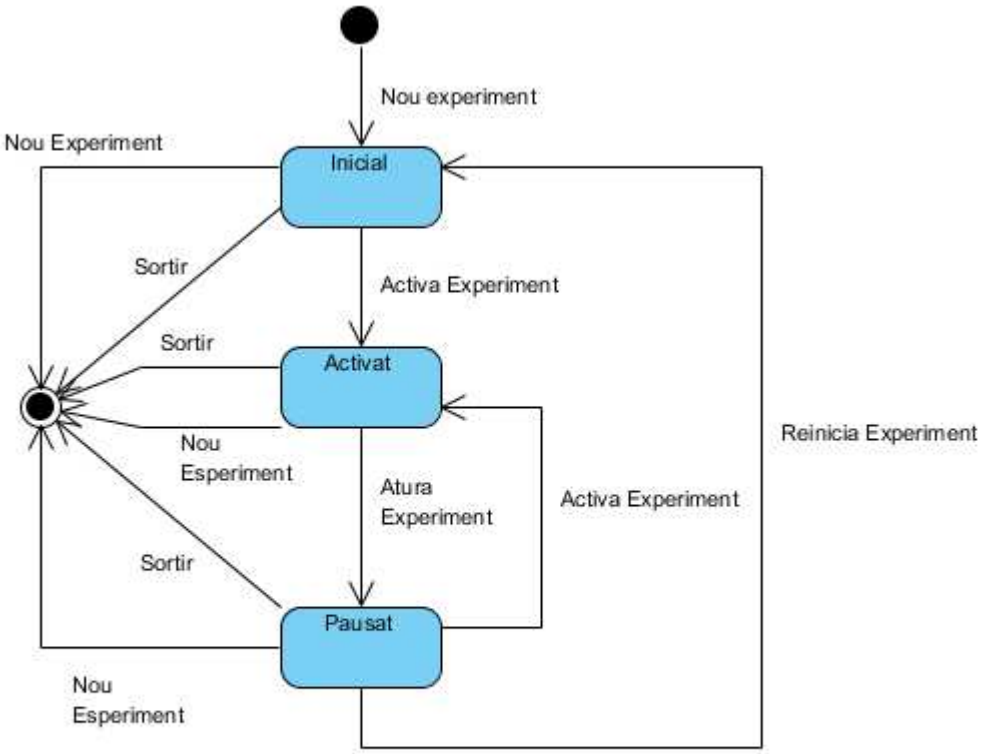
Usuari	Sistema
5- L'usuari fa clic en "Exportar dades a Excel"	6- El sistema mostra una finestra amb l'opció de el nom i la ruta del fitxer a generar.
7- L'usuari introdueix la ruta i el nom del fitxer.	9- El sistema genera a la ruta indicada amb el nom que ha introduït l'usuari.

3.2 Especificació estats

Els diagrames d'estats són una tècnica coneguda per descriure el comportament d'un sistema. Descriuen tots els estats possibles en els quals pot entrar un sistema particular i la manera en què canvia l'estat de l'objecte, com a resultat dels esdeveniments que arriben a ell. En la major part de les tècniques orientades a objectes, els diagrames d'estats es dibuixen per a una sola classe, mostrant el comportament d'un sol objecte durant tot el seu cicle de vida.

3.2.1 Diagrama d’estats

En la següent figura s’observa el comportament del sistema i les possibles transicions d’estats.

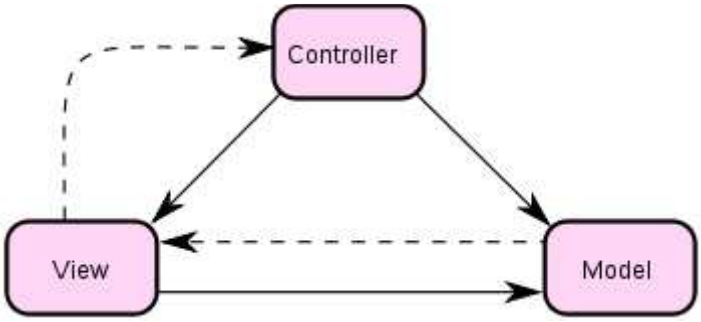


Il·lustració 33 Diagrama d’estats

3.3 Model de disseny

3.3.1 Patró de disseny principal: Model vista controlador

El Model Vista Controlador (MVC) és un patró d'arquitectura de software que separa les dades d'una aplicació, la interfície d'usuari, i la lògica de control en tres components diferents. El patró de trucada i tornada MVC, es veu freqüentment en aplicacions web, on la vista és la pàgina HTML i el codi que proveeix de dades dinàmiques a la pàgina. El model és el Sistema de Gestió de Base de dades i la Lògica de negoci, i el controlador és el responsable de rebre els esdeveniments d'entrada des de la vista.

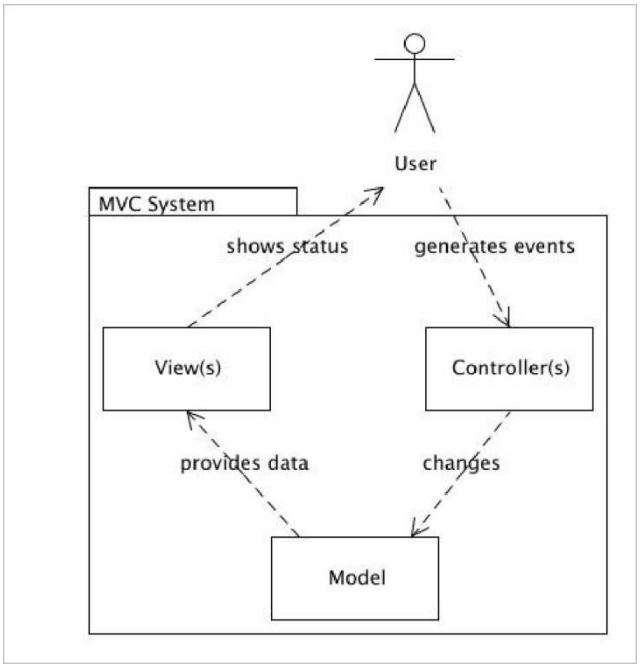


Il·lustració 34 Patró de disseny MVC

3.3.2 Descripció del patró

- **Model:** Aquesta és la representació específica de la informació amb la qual el sistema opera. En resum, el model es limita al relatiu de la vista i el seu controlador facilitant les presentacions visuals complexes. El sistema també pot operar amb més dades no relatius a la presentació, fent ús integrat d'altres lògiques de negoci i de dades afins amb el sistema modelat.
- **Vista:** Representa el model en un format adequat per interactuar, usualment la interfície d'usuari.
- **Controlador:** Es el que respon a esdeveniments, usualment accions de l'usuari, i invoca peticions al model i, probablement, a la vista.

Molts dels sistemes informàtics utilitzen un Sistema de Gestió de Base de dades per gestionar les dades: en línies generals del MVC correspon al model. La unió entre capa de presentació i capa de negoci conegut en el paradigma de la programació per capes representaria la integració entre Vista i el seu corresponent Controlador d'esdeveniments i accés a dades, MVC no pretén discriminar entre capa de negoci i capa de presentació però si pretén separar la capa visual gràfica de la seva corresponent programació i accés a dades, alguna cosa que millora el desenvolupament i manteniment de la Vista i el Controlador en paral·lel, ja que tots dos compleixen cicles de vida molt diferents entre si.



Il·lustració 35 MVC amb l’usuari

Encara que es poden trobar diferents implementacions de MVC, el flux que segueix el control generalment és el següent:

1. L'usuari interactua amb la interfície d'usuari d'alguna forma (per exemple, l'usuari prem un botó, enllaç, etc.)

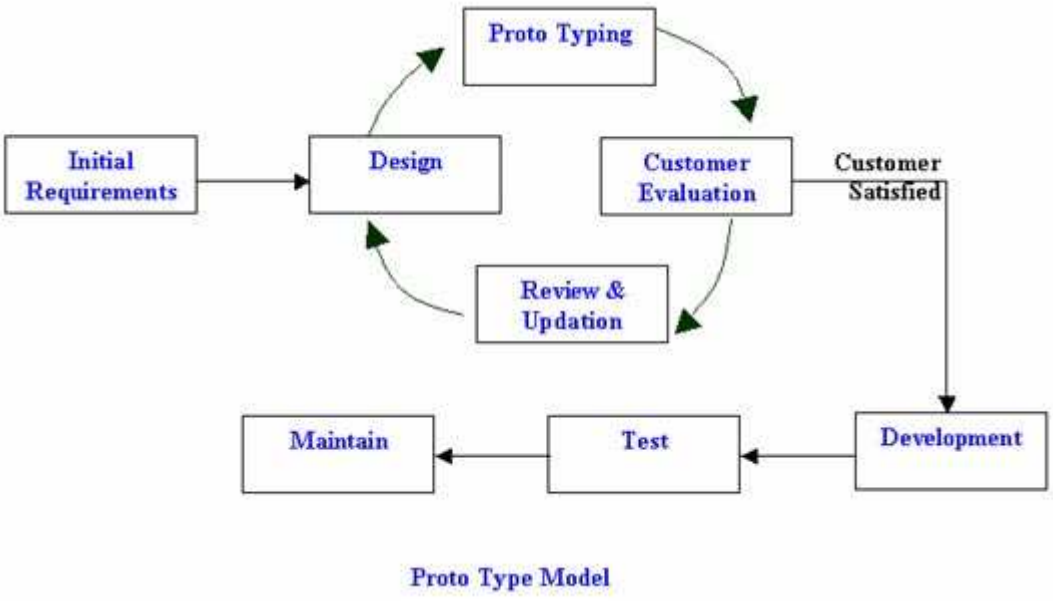
2. El controlador rep (per part dels objectes de la interfície - vista) la notificació de l'acció sol·licitada per l'usuari. El controlador gestiona l'esdeveniment que arriba, freqüentment a través d'un gestor d'esdeveniments (handler) o callback.
3. El controlador accedeix al model, actualitzant-ho, possiblement modificant-ho de forma adequada a l'acció sol·licitada per l'usuari (per exemple, el controlador actualitza el carro de la compra de l'usuari). Els controladors complexos estan sovint estructurats usant un patró de comando que encapsula les accions i simplifica la seva extensió.
4. El controlador delega als objectes de la vista la tasca de desplegar la interfície d'usuari. La vista obté les seves dades del model per generar la interfície apropiada per a l'usuari on es reflecteixen els canvis en el model (per exemple, produeix un llistat del contingut del carro de la compra). El model no ha de tenir coneixement directe sobre la vista. No obstant això, es podria utilitzar el patró Observador per proveir certa in direcció entre el model i la vista, permetent al model notificar als interessats de qualsevol canvi. Un objecte de la vista pot registrar-se amb el model i esperar als canvis, però així i tot el model en si mateix segueix sense saber gens de la vista.
5. El controlador no passa objectes de domini (el model) a la vista encara que pot donar l'ordre a la vista perquè s'actualitzi. Nota: En algunes implementacions la vista no té accés directe al model, deixant que el controlador enviï les dades del model a la vista.
6. La interfície d'usuari espera noves interaccions de l'usuari, començant el cicle novament.

3.4 Model de desenvolupament

3.4.1 Prototip : desenvolupament evolutiu

En Enginyeria de programari el model de prototips que pertany als models de desenvolupament evolutiu, el prototip ha de ser construït en poc temps, usant els programes adequats i no s'ha d'utilitzar molts diners doncs a partir de que aquest sigui aprovat nosaltres podem iniciar el veritable desenvolupament del programari.

El disseny ràpid es centra en una representació d'aquells aspectes del programari que seran visibles per al client o l'usuari final. Aquest disseny condueix a la construcció d'un prototip, el qual és avaluat pel client per a una retroalimentació; gràcies a aquesta es refinen els requisits del programari que es desenvoluparà. La interacció succeeix quan el prototip s'ajusta per satisfer les necessitats del client. Això permet que al mateix temps el desenvolupador entengui millor el que s'ha de fer i el client vegi resultats a curt termini.



Il·lustració 36 Model de prototipus

Etapes:

1. Pla ràpid: Prendre requeriments i analitzar viabilitat.
2. Modelatge, disseny ràpid: Anàlisis i disseny del programa.
3. Construcció del Prototip: Implementació del programa
4. Desenvolupament, lliurament i retroalimentació: Fase iterativa on es desenvolupa el programa s'entrega al client i es fan les correccions per perfeccionar el prototipus.
5. Comunicació: Reunions conjuntes on s'ensenya el programa al client.

Avantatges

- Aquest model és útil quan el client coneix els objectius generals per al software, però no identifica els requisits detallats d'entrada, processament o sortida.
- També ofereix un millor enfocament quan el responsable del desenvolupament del programari està insegur de l'eficàcia d'un algorisme, de l'adaptabilitat d'un sistema operatiu o de la forma que hauria de prendre la interacció usuari-sistema.

La construcció de prototipus es pot utilitzar com un model del procés independent, s'empra més comunament com una tècnica susceptible d'implementar-se dins del context de qualsevol dels models del procés exposats. Sense importar la forma en què aquest s'apliqui, el paradigma de construcció de prototips ajuda al desenvolupador de software i al client a entendre de millor manera quin serà el resultat de la construcció quan els requisits estiguin satisfets. D'aquesta manera, aquest cicle de vida en particular, involucra al client més profundament per adquirir el producte.

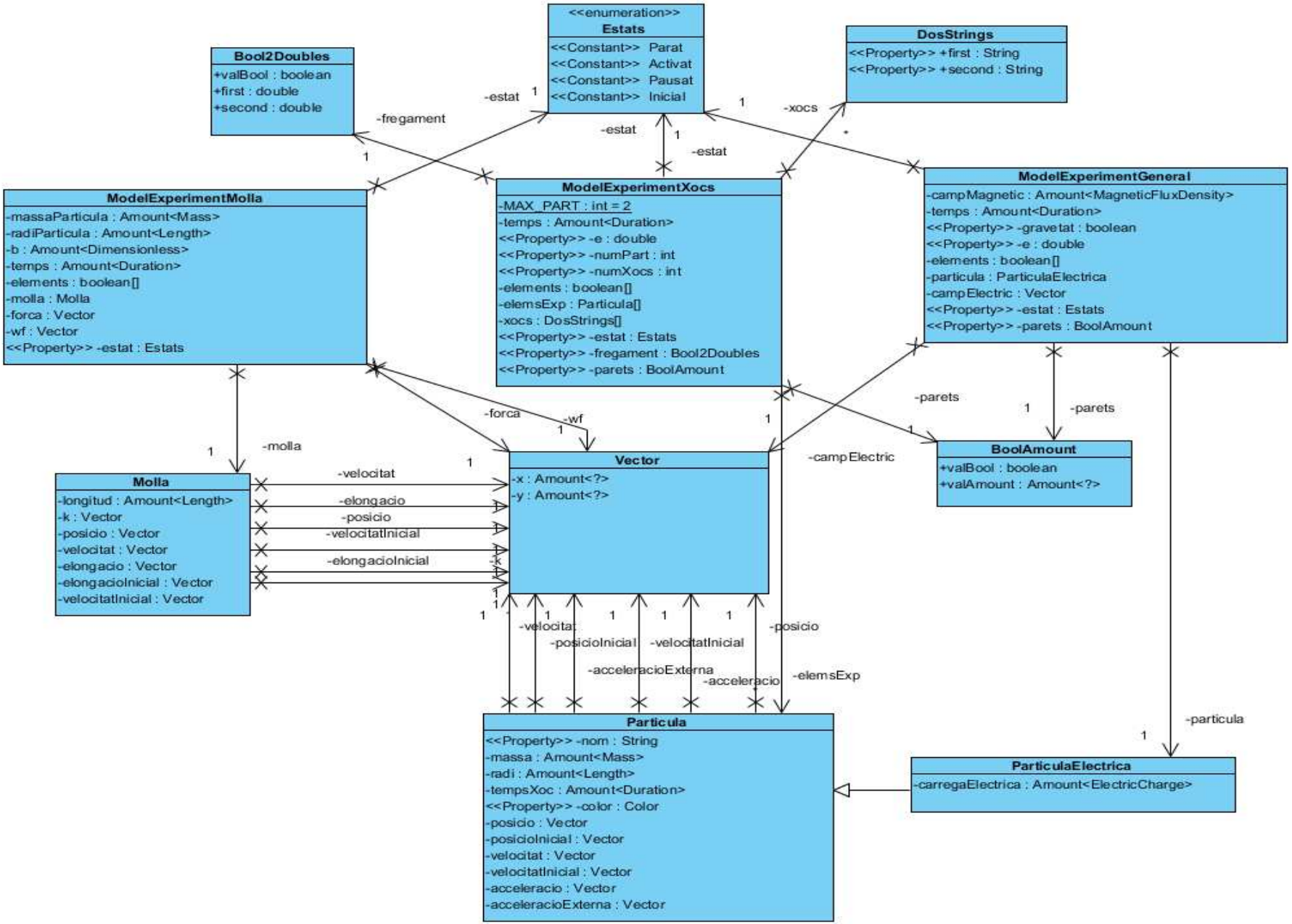
Inconvenients

- L'usuari tendeix a crear-se unes expectatives quan veu el prototip de cara al sistema final. A causa de la intenció de crear un prototipus de forma ràpida, se solen desatendre aspectes importants, tals com la qualitat i el manteniment a llarg termini, la qual cosa obliga en la major part dels casos a reconstruir-ho una vegada que el prototip ha complit la seua funció. És freqüent que l'usuari es mostri poc inclinat i demana que sobre aqueix prototip es construeixi el sistema final, la qual cosa ho convertiria en un prototip evolutiu, però partint d'un estat poc recomanat.

- Per desenvolupar ràpidament el prototipus, el desenvolupador sol prendre algunes decisions d'implementació poc convenients (per exemple, triar un llenguatge de programació incorrecte perquè proporciona un desenvolupament més ràpid). Amb el pas del temps, el desenvolupador pot oblidar-se de la raó que li va portar a prendre tals decisions, amb el que es corre el risc que aquestes eleccions passen a formar part del sistema final.

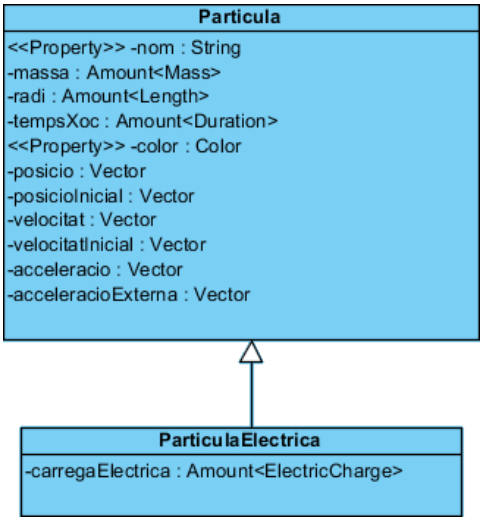
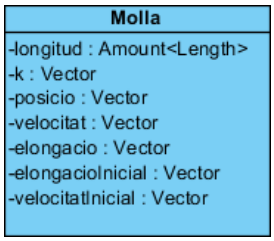
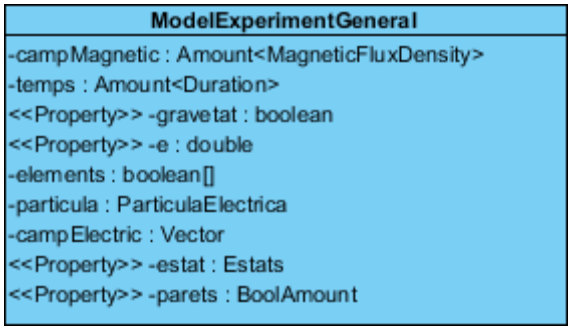
3.5 Diagrama de classes:

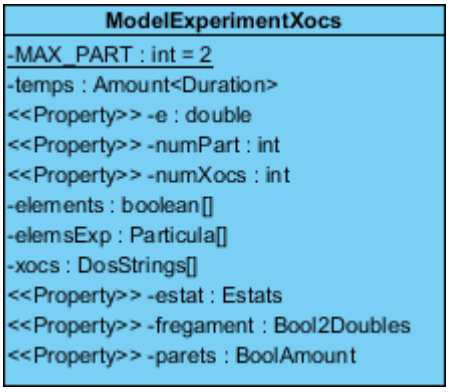
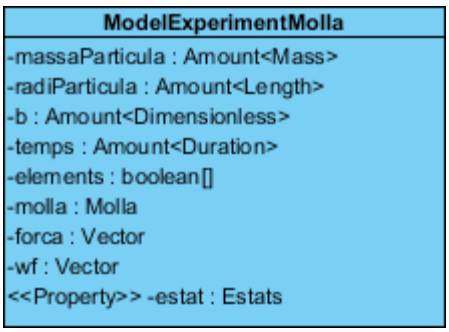
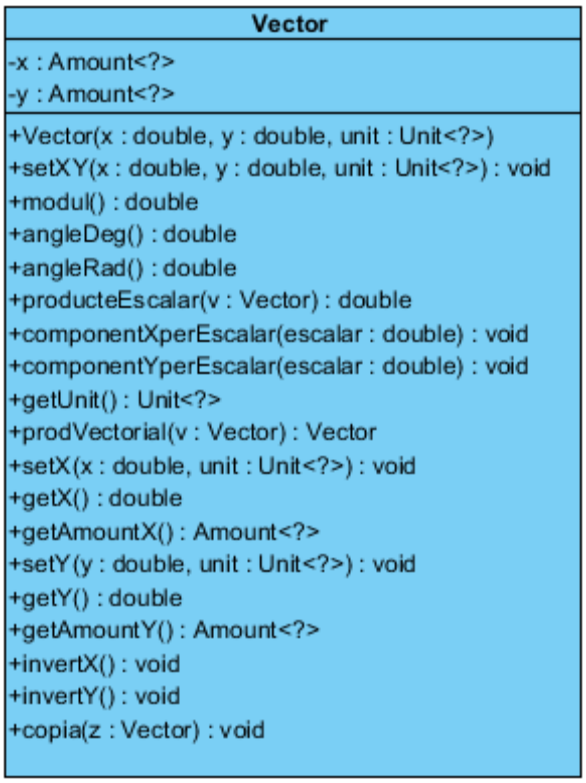
Un diagrama de classes és un tipus de diagrama estàtic que descriu l'estructura d'un sistema mostrant les seves classes, atributs i les relacions entre ells. Els diagrames de classes són utilitzats durant el procés d'anàlisi i disseny dels sistemes, a on es crea el disseny conceptual de la informació que utilitzarà el sistema, i els components que s'encarregaran del seu funcionament i de la relació entre l'un i l'altre.

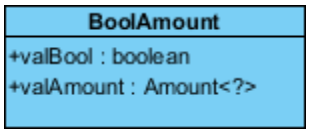
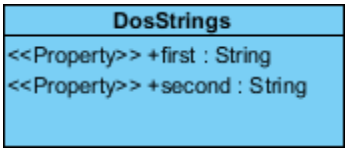
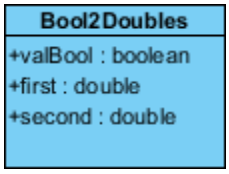


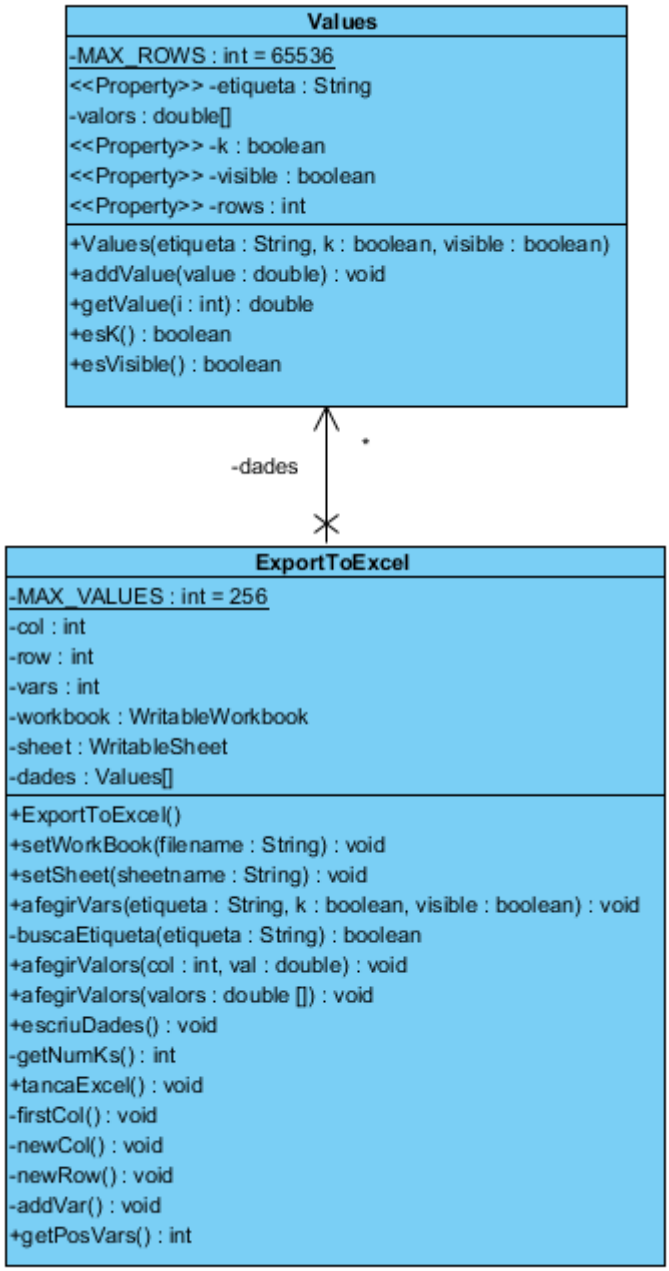
Il·lustració 37 Diagrama de classes

3.5.1 Model

Classe	Explicació
 Il·lustració 38 Classe Partícula	Una partícula representa l'element a estudiar en els experiments. En ella es registren totes les dades necessàries com per exemple radi, velocitat, posició etc… Existeix una concreció d'una partícula, aquella que te propietats elèctriques, es per això que es representa en un model d'herència.
 Il·lustració 39 Classe Molla	Una molla es la representació d'un element elàstic amb una sèrie de propietats que s'enregistren per poder tenir l'estat en qualsevol moment de l'experiment.
 Il·lustració 40 Classe ModelExperimentGeneral	El model on es representarà els experiments generals. En ell es guarda l'estat de l'experiment, el temps la partícula i les propietats físiques que intervenen.

Clase	Explicació
 Il·lustració 41 Classe ModelExperimentXocs	El model on es representarà els experiments generals. En ell es guarda l'estat de l'experiment, el temps, les partícules i les propietats físiques que intervenen.
 Il·lustració 42 Classe ModelExperimentXocs	El model on es representarà els experiments generals. En ell es guarda l'estat de l'experiment, el temps, les molla i les propietats físiques que intervenen.
	Representar un vector es necessari, ja que cal representar totes les magnituds vectorials als experiments (que n'hi ha moltes). En aquesta classe s'han definit les diferents operacions possibles entre vectors i elements escalars, facilitant les operacions necessàries.

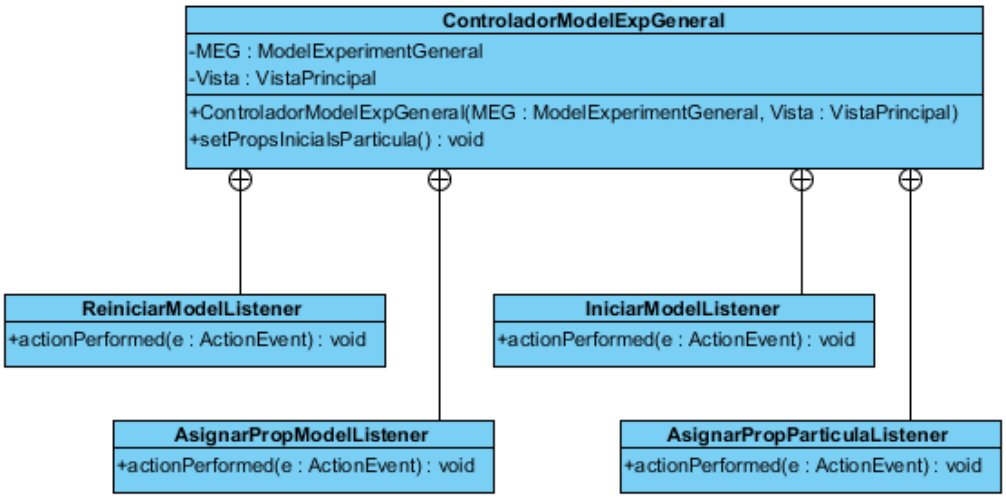
Il·lustració 43 Classe Vector	
Clase	Explicació
 Il·lustració 44 Classe BoolAmount	Es la classe que representa les parets, es la combinació d'un valor booleà i una quantitat.
 Il·lustració 45 Classe DosStrings	Aquesta classe representa el nom de les partícules que col·lisionen.
 Il·lustració 46 Classe Bool2Double	Aquesta classe representa el nom de les partícules que col·lisionen, es necessari per identificar-les.

Clase	Explicació
 Il·lustració 47 Classes Exportació a Excel	Aquestes dos classe s'utilitzen per desar els valors generats en els experiments per després poder-ho exportar a una fulla Excel. També conte la representació de la fulla de càlcul per modificar les propietats del la fulla de càlcul.

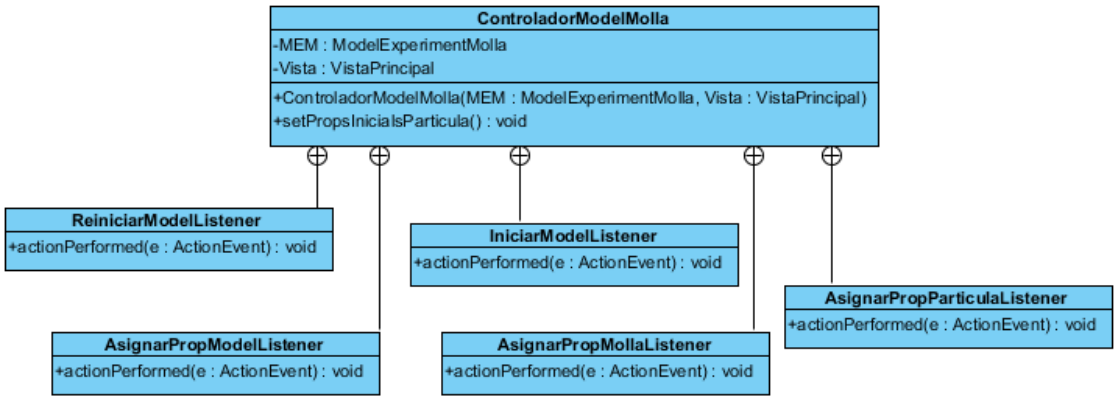
3.5.2 Controlador

Aquests controladors s’encarreguen de comunicar vista i model, s’encarreguen de capturar els esdeveniments produïts per la vista i els transmet al model, una cop el model els ha processa’t el controlador actualitza la vista amb el resultat. Les accions que realitzen els controladors coincideixen amb els casos d’us explicats anteriorment.

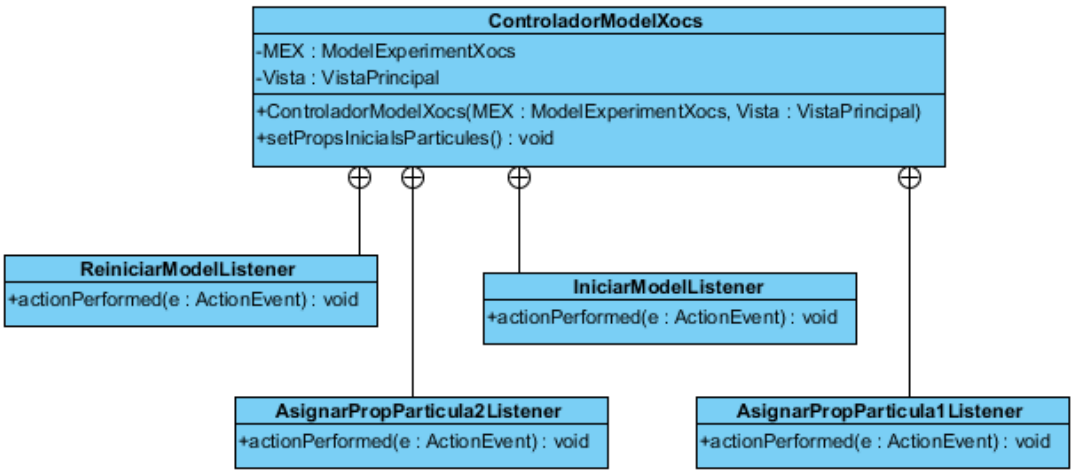
Els tres primers controladors son un per cada model d’experiment, i el darrer es el que s’encarrega de gestionar els esdeveniments relacionats amb l’exportació de dades i la gestió de la serialització.



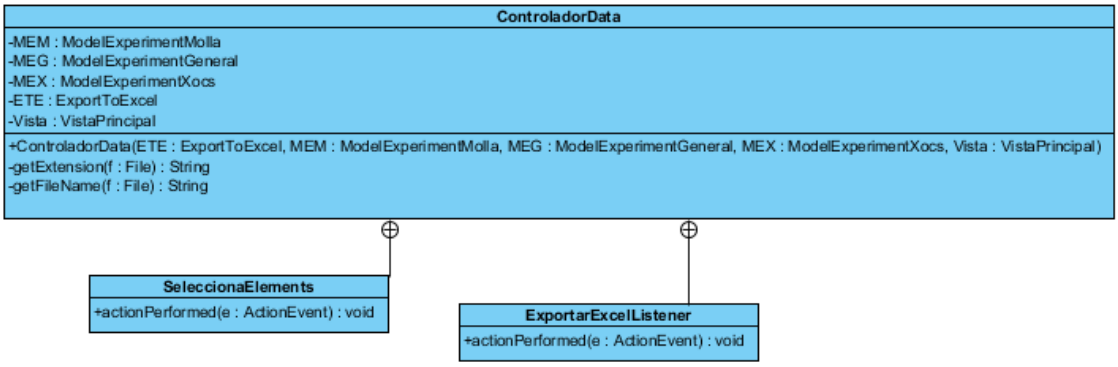
Il·lustració 48 Controlador Experiment General



Il·lustració 49 Controlador ExperimentMolla

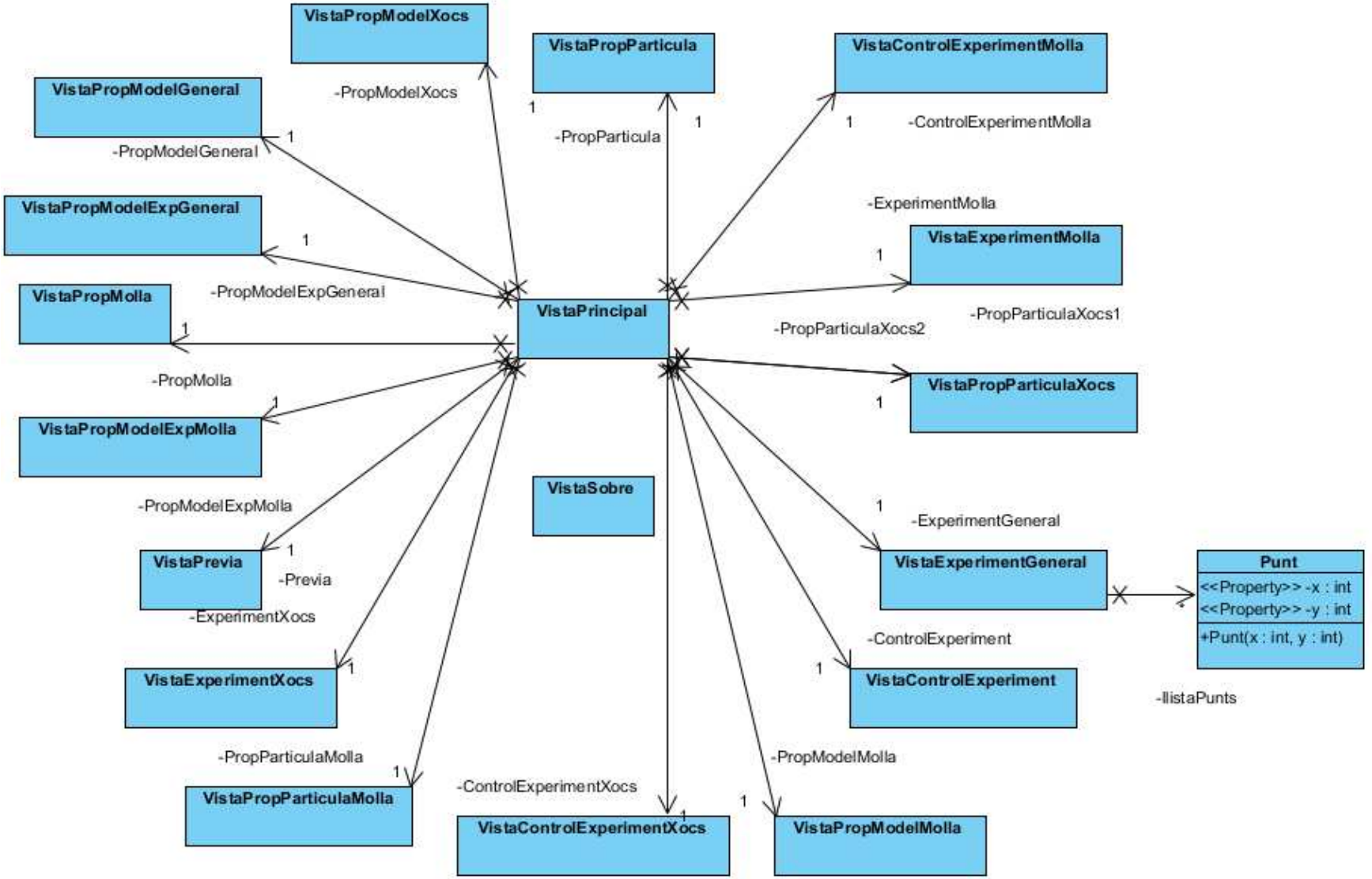


Il·lustració 50 Controlador Experiment Xocs



Il·lustració 51 Controlador exportació i serialització de dades

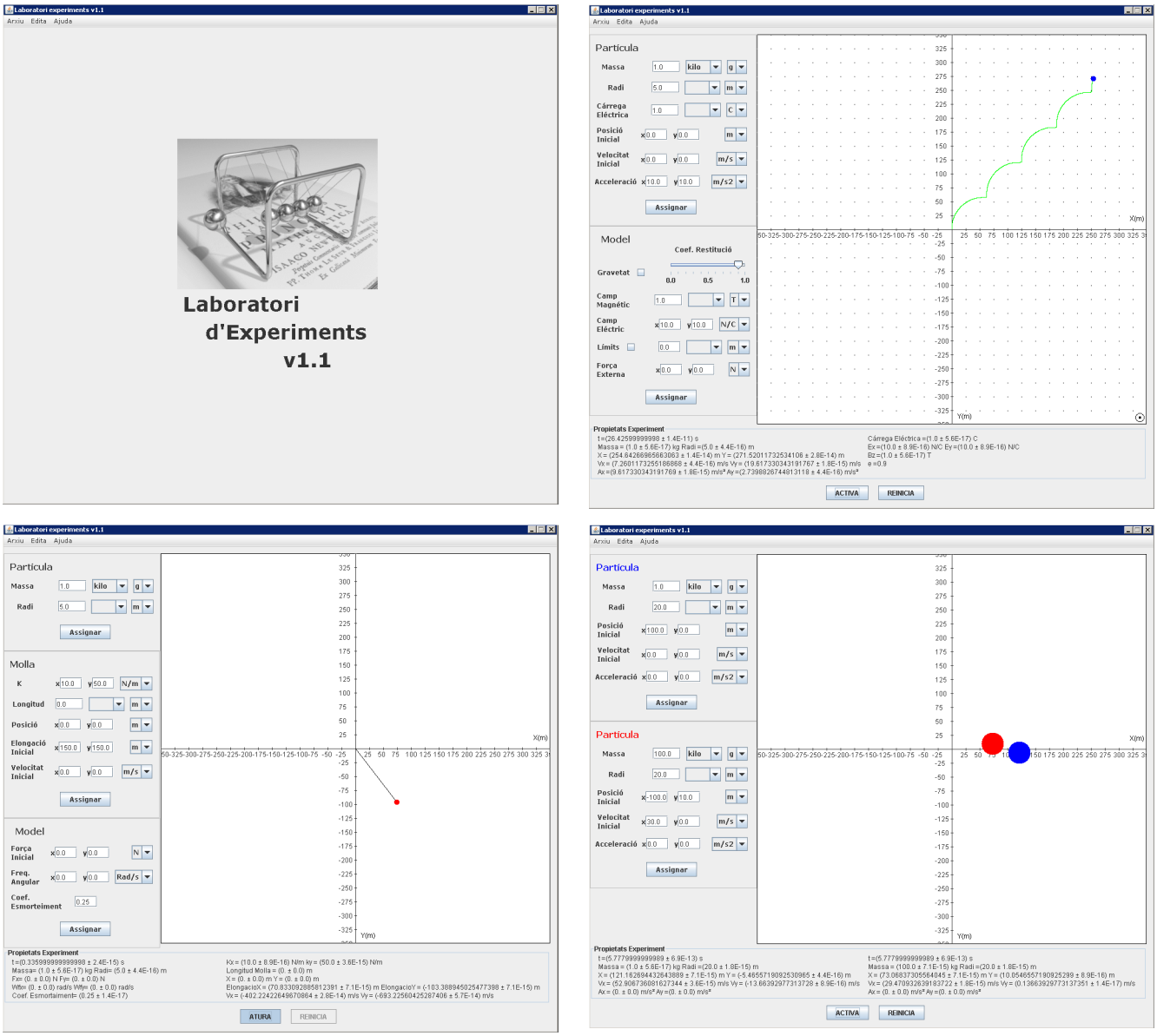
3.5.3 Vistes



Il·lustració 52 Classes de la capa de presentacions

4 Disseny de l’ interfície

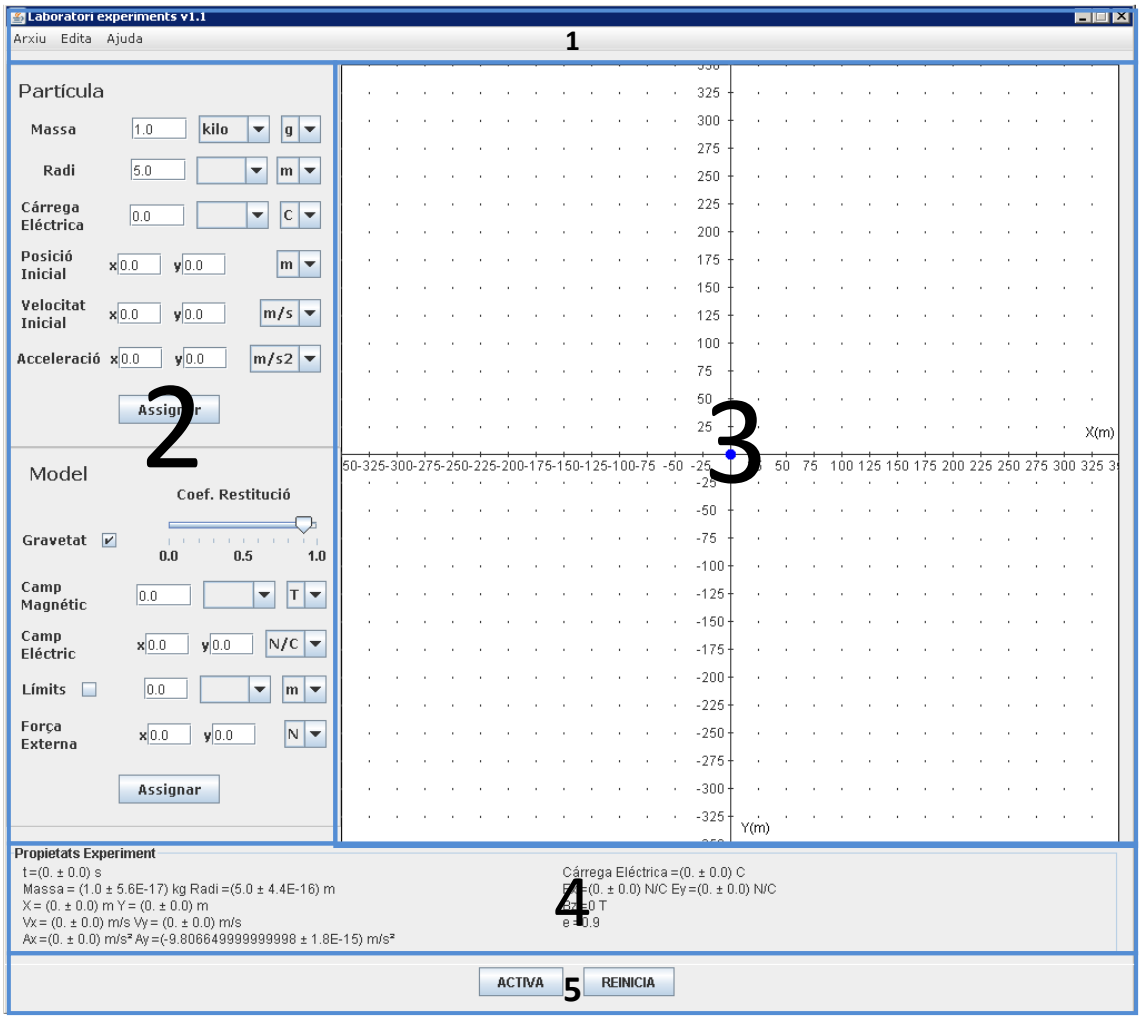
En aquest apartat es detalla el disseny extern de l'aplicació, el que l'usuari podrà veure. Per al disseny de la interfície s'han tingut en compte els requisits no funcionals com claredat, simplicitat i usabilitat. Pel que se segueixen els criteris de bon disseny i estructuració lògica d'elements.



Il·lustració 53 Aspecte de l’aplicació

4.1 Estructura general de la interfície

A continuació es mostra com estan estructurades les pantalles de l'aplicació. Aquesta estructura es comuna a tots els models d'experiments, facilitant la utilització de l'aplicació i s'entengui més ràpidament com funciona. També ha estat pensat el disseny per poder re-aprofitar-ho si s'afegeixen nous models d'experiments.



1 **Menú principal**: És la part comuna de tota l'aplicació, en el menú estan les opcions de programa que estan actives en funció de l'execució del programa.

2 **Panell per modificar propietats**: Aquí es troben els apartats específics de cada experiment, en funció de l'experiment actiu mostra els elements a modificar.

3 **Panell experiment**: El panell on es mostra l'acció dels experiments, es on es representen les dades generades en les fórmules que es processen.

4 **Panell propietats components del experiment**: Es on es veu en directe l’evolució dels elements de l’experiment. En funció de l’experiment es mostren uns elements o d’altres. Es mostren tant constants com elements variables, el temps serà un element variable o per exemple la massa serà constant (cal dir que, durant l’execució del experiment no es pot alterar les propietats inicials, però si s’atura es poden modificar).

5 **Panell control d’estat del experiment**: Amb aquest panell es controla l’estat del experiment. Els estats que pot tenir i com pot arribar s’expliquen en el diagrama d’estats del sistema.

5 Eines utilitzades per realitzar el projecte

5.1 JScience[xv]

Els desenvolupadors Java que treballen amb quantitats físiques (com desenvolupadors en el camp científic, enginyer, mèdic etc..) han de ser capaços de tenir mesures d'aquestes quantitats en els seus programes. Els models inadequats de magnituds físiques poden dirigir a errors significatius. En particular, a la pràctica de modelat, una mesura com a número senzill sense considerar les unitats, pot representar la creació de codi fràgil. Un altre desenvolupador o una altra part del codi pot interpretar erròniament un número, mentre aquest representa una unitat diferent de mesura. Per exemple, pot succeir que el pes d'una persona és expressat en lliures, quilograms, tones etc..

Els desenvolupadors tampoc han d'utilitzar models inadequats de mesura, o ha de crear les seves solucions pròpies. Una solució comuna pot ser més segura i pot salvar temps de desenvolupament, o fer-ho de manera específica. Aquest JSR[11] proposa per establir mètodes segurs i útils per modelat quantitats físiques.

[11] El Procés de la Comunitat Java, o Java Community Process, establert en 1998, és un procés formalitzat el qual permet a les parts interessades a involucrar-se en la definició de futures versions i característiques de la plataforma Java.

El procés JCP comporta l'ús de Java Specification Request (JSR), les quals són documents formals que descriuen les especificacions i tecnologies proposades perquè siguin afegides a la plataforma Java. Les revisions publiques formals de JSRs són controlades abans que els *SR es converteixin en final i siguin votats pel Comitè Executiu JCP. Un JSR final subministra una implementació de referència la qual dóna una implementació lliure de la tecnologia en codi font i un Kit de Compatibilitat de Tecnologia per verificar l'especificació de la API.

El JCP mateix està descrit per un JSR. Des de 2006, la versió actual de JCP en ús és la 2.6 com es descriu per JSR 215.

A la fi del 2008, el JCP està compost per més de 1200 membres, tots ells empreses capdavanteres en el sector tecnològic i del desenvolupament de programari. Hi ha aproximadament 300 JSR.

JSR-275 especifica un o més paquets Java per que el programador pugui tenir quantitats físiques i la seva expressió com números d'unitats. L'especificació inclou:

- Comprovació de comptabilitat d'unitats a les expressions

- Expressar les mesures o elements en diferents unitats.

- Operacions aritmètiques en les unitats

- Classes per transformar i formatatge de representacions d'unitats textuals.

- Les classes Concretes que implementen els tipus estàndards d'unitats (com les base i les unitats derivades) i conversions d'unitat.

- Una base de dades de unitats predefinides.

5.1.1 Exemple d'utilització:

Package org.jscience.physics.amount

Proporciona suport per precisió exacta o arbitrària de les mesures.

`Amount<Q extends Quantity>` com a mesura de quantitats

```
// Mesures Exactes.

Amount<Mass> m0 = Amount.valueOf(100, POUND); // Representació d'un enter

Amount<Mass> m1 = m0.times(33).divide(2); // Continua com un enter.

// Conversions exactes.

Amount<ElectricCurrent> m2 = Amount.valueOf("234 mA").to(MICRO(AMPERE));

> m0 = 100 lb
> m1 = 1650 lb
> m2 = 234000 µA

// Mesures no exactes.

Amount<Mass> m3 = Amount.valueOf(100.0, POUND);
// IEEE 754 64-bits float precisió.

Amount<Mass> m4 = m0.divide(3); // Representació d'un enter inexacte.

Amount<ElectricCurrent> m5 = Amount.valueOf("234 mA").to(AMPERE);
// Conversió introdueix errors.

Amount<Temperature> t0 = Amount.valueOf(-7.3, 0.5, DEGREE_CELSIUS);
// Error explicit.

> m3 = (100.0 ± 2.1E-14) lb
> m4 = (33.333333333333336 ± 7.1E-15) lb
> m5 = (2.3400000000000004E-1 ± 4.2E-17) A
```

```
> t0 = (-7.30 ± 5.0E-1) °C


// Mesures d’intervals
Amount<Volume> m6 = Amount.valueOf(20, 0.1, LITER);
Amount<Frequency> m7 = Amount.rangeOf(10, 11, KILO(HERTZ));

> m6 = (20.00 ± 1.0E-1) L
> m7 = (10.5 ± 5.0E-1) kHz

// Amount.equals (identics)
// Amount.approximates (aproximat)

Amount<Frequency> m8 = Amount.valueOf(9000, HERTZ);
Amount<Frequency> m10 = m8.divide(3).times(3); // Continua exacte.
Amount<Frequency> m11 = m8.divide(7).times(7); // Ja no és exacte.

System.out.println("m8 = " + m8);
System.out.println("m10 = " + m10);
System.out.println("m11 = " + m11);
System.out.println("(m10 == m8) = " + m10.equals(m8));
System.out.println("(m10 ≅ m8) = " + m10.approximates(m8));
System.out.println("(m11 == m8) = " + m11.equals(m8));

> m8 = 9000 Hz
> m10 = 9000 Hz
> m11 = (9000.0 ± 1.8E-12) Hz
> (m10 == m8) = true
> (m10 ≅ m8) = true
> (m11 == m8) = false
> (m11 ≅ m8) = true

// AmountFormat - Multiplicació/Resta Error (3 dígits d’error)

AmountFormat.setInstance(AmountFormat.getPlusMinusError(3));

> m3 = (100.0 ± 2.13E-14) lb
> m4 = (33.33333333333333504 ± 7.11E-15) lb
> m5 = (2.340000000000000512E-1 ± 4.16E-17) A

// AmountFormat – Error de suport (2 digits error)

AmountFormat.setInstance(AmountFormat.getBracketErrorInstance(2));

> m3 = 100.0000000000000000[21] lb
> m4 = 33.33333333333333376[71] lb
> m5 = 2.34000000000000032[41]E-1 A

// AmountFormat – Nomes dígits exactes

AmountFormat.setInstance(AmountFormat.getExactDigitsInstance());

> m3 = 100.000000000000 lb
> m4 = 33.3333333333333 lb
> m5 = 2.340000000000000E-1 A
```

```
// Errors numerics

Amount<Length> x = Amount.valueOf(1.0, METRE);
Amount<Velocity> v = Amount.valueOf(0.01, METRE_PER_SECOND);
Amount<Duration> t = Amount.valueOf(1.0, MICRO(SECOND));

for (int i = 0; i < 10000000; i++) {
        x = x.plus(v.times(t));
}

AmountFormat.setInstance(AmountFormat.getExactDigitsInstance());

> x = 1.1000000 m   (esta a l'interval [1.0999999, 1.1000001])

// Camparació amb tipus double.

double x = 1.0; // m
double v = 0.01; // m/s
double t = 1E-6; // s

for (int i = 0; i < 10000000; i++) {
        x += v * t;
}

> x = 1.099999999392253 (no sabem la precisió)
```

5.2 Java Excel APIxvi

Java Excel API és una API[12] per Java de codi obert que habilita a desenvolupadors per llegir, escriu, i modificar Excels dinàmicament. Ara els desenvolupadors poden llegir Excels, els modifica amb un API convenient i senzill, i escriure els canvis a qualsevol tipus de sortida (p. ex. disc, HTTP, base de dades, o qualsevol casquet).

Qualsevol sistema operatiu que pot córrer amb un MVJ[13], pot ambdós processar i repartir Excels. Perquè és Java, els API poden ser invocats de dins d'un servlet, es per això que dona accés per Internet i aplicacions de web d’intranets.

Algunes Característiques:

- Llegeix dades de llibres Excel 95, 97, 2000, XP, i 2003
- Llegeix i escriu fórmules (Excel 97 i versions posteriors)
- Genera fulles de càlcul en Excel 2000
- Suporta canviar format font, número i formatatge de les dades.
- Dóna suport en ombrejar, modificar vores, i donar color a les celes
  Modifica llibres de càlcul existents
- És multi llenguatge, habilitant processament dins gairebé qualsevol país, llengua, o el caràcter que codifica (les fórmules actualment només donen

[12] Una interfície de programació d'aplicacions o API (de l'anglès Application Programming Interface) és el conjunt de funcions i procediments (o mètodes, a la programació orientada a objectes) que ofereix certa biblioteca per ser utilitzat per un altre programari com una capa d'abstracció.

[13] Una Màquina Virtual Java (MVJ) és un conjunt de programes d'ordinador i estructures de dades que implementen un model específic de màquina virtual. Aquest model accepta un tipus de llenguatge intermedi, anomenat normalment amb bytecode, conceptualitzant aquest representa el conjunt d'instruccions d'un llenguatge de Programació basat en pila i una capa d'arquitectura orientada a la seguretat. Aquest codi majoritàriament està generat pels compiladors del Llenguatge Java, encara que la MVJ també pot estar construïda per compiladors d'altres llenguatges. La MVJ és un component crucial de la Plataforma Java. A causa del fet que les MVJ estan disponibles per moltes plataformes de maquinari i de programari, Java, tant pot ser un middleware i una plataforma de ple dret -d'aquí l'expressió escriu un cop, executa arreu, La utilització del mateix bytecode per a totes les plataformes, permet a Java descriure's com "escriu un cop, executa arreu", en contraposició a l'"escriu un cop, compila arreu". La MVJ permet funcionalitats úniques, com la Gestió automàtica d'excepcions , que proporciona informació de depuració per a qualsevol error en el programari (excepcions) independents del codi font.

suport en anglès, francès, espanyol, i alemany, però es poden afegir si es tradueixen)

- Dóna suport per copiar gràfics
- Dóna suport per inserir i copiar imatges a les fulles de càlcul

5.2.1 Exemple d'utilització:

```
// Llegir d'una fulla de càlcul.

import java.io.File;
import java.util.Date;
import jxl.*;

// Creem una nova fulla de càlcul sobre un nou arxiu
Workbook workbook = Workbook.getWorkbook(new File("myfile.xls"));

// Seleccionem la primera pestanya
Sheet sheet = workbook.getSheet(0);

// Creem tres celes
Cell a1 = sheet.getCell(0,0);
Cell b2 = sheet.getCell(1,1);
Cell c2 = sheet.getCell(2,1);

// Obtenim el contingut
String stringa1 = a1.getContents();
String stringb2 = b2.getContents();
String stringc2 = c2.getContents();

// Creem tres variables amb els diferents tipus bàsics
String stringa1 = null;
double numberb2 = 0;
Date datec2 = null;

// Creem tres celes
Cell a1 = sheet.getCell(0,0);
Cell b2 = sheet.getCell(1,1);
Cell c2 = sheet.getCell(2,1);

// Comparem el tipus de celes i fem l'assignació correcta
if (a1.getType() == CellType.LABEL)
{
  LabelCell lc = (LabelCell) a1;
  stringa1 = lc.getString();
}

if (b2.getType() == CellType.NUMBER)
{
  NumberCell nc = (NumberCell) b2;
  numberb2 = nc.getValue();
}
```

```
if (c2.getType() == CellType.DATE)
{
  DateCell dc = (DateCell) c2;
  datec2 = dc.getDate();
}

// Escriure en una fulla de càlcul.

import java.io.File;
import java.util.Date;
import jxl.*;
import jxl.write.*;

// Creem un nou llibre sobre un nou arxiu
WritableWorkbook workbook = Workbook.createWorkbook(new File("output.xls"));

// Creem una nova fulla de càlcul per escriure
WritableSheet sheet = workbook.createSheet("First Sheet", 0);

// Posem una capçalera
Label label = new Label(0, 2, "A label record");
sheet.addCell(label);

// Afegim valors numèrics
Number number = new Number(3, 4, 3.1459);
sheet.addCell(number

// Desem el que hem escrit i tanquem el llibre
workbook.write();
workbook.close();
```

5.3 JNLP[xvii]

JNLP és l'acrònim de Java Networking Launching Protocol que és una especificació usada per Java Web Start. Aquesta especificació, permet tenir centralitzat en un servidor web un programa, evitant els problemes de distribució i instal·lació.

En instal·lar qualsevol aplicació, normalment es donen una sèrie de passos:

- Es descarrega d'internet/ s'introdueix un mitjà extraïble (CD/DVD/Disc USB...)
- S'instal·la en l'ordinador
- S'executa

El programa **javaws** permet fer això d'una manera més fàcil i transparent per a l'usuari, ja que podem realitzar aquests passos simplement punxant sobre un enllaç mentre estem utilitzant el nostre navegador, de manera que descàrrega, instal·lació i execució es realitzen de manera transparent a l'usuari. JavaWS no usa applets, ja que descarrega aplicacions java normals i necessita, per tant, d'una màquina virtual, a més, ve inclòs en el JRE de java des de la versió 1.4.

Qualsevol enllaç JNLP, en iniciar el procés d'execució, demana autorització a l'usuari. A més, les aplicacions poden estar signades (signatura electrònica) per assegurar el remitent de l'aplicació de manera que poden seguir el model de seguretat de la plataforma Java 2 per assegurar la integritat de les dades que obtenim a través de la xarxa, de manera que no es produeixin atacs de tipus *Man in the Middle*, *DNS poisoning*, o corrupció de dades.

5.3.1 Exemple d’utilització:

Un arxiu JNLP és un XML especialment format composat per:

- Una capçalera XML típica:

  ```
  <?xml version="1.0" encoding="CODIFICACIÓ"?>
  ```

  On codificació pot ser qualsevol codificació vàlida: utf-8, ISO-8859-1...
- Una ruta predeterminada perquè els arxius puguin ser cridats des d'un path relatiu.

<jnlp spec="1.0+" codebase="http://URL/directorio/del/programa" href="NombreDelArchivoJNLP.jnlp">

- Un o diversos TAG information en el qual van diverses informacions (veure exemple).
- Un TAG security (amb informació variada).
- Un TAG resources (amb informació variada).
- Un TAG application-desc amb la classe predeterminada a executar.

El següent és un breu exemple d'un arxiu típic JNLP. No inclou totes les possibles opcions. Hi ha moltes més que poden veure's des de la documentació de Java.

```
<?xml version="1.0" encoding="utf-8"?>
<jnlp spec="1.0+"
codebase="http://es.wikipedia.org/wiki/JNLP:8080/"
href="jnlp.jnlp">

<information>
   <title>Ejemplo de un JNLP wikipédico</title>
   <vendor>Anónimo</vendor>
   <homepage href="http://es.wikipedia.org/Portada" />
        <description> Ejemplo de un JNLP muy
wikipédico</description>
        <description kind="short">
               Esta es una breve información
        </description>
    <icon href="NombreImagen.jpg" />
    <offline-allowed />
</information>

<security>
    <all-permissions />
</security>

<resources>
   <j2se version="1.4+" />
   <jar href="aplicación.jar" />
   <jar href="lib1.jar" />
   <jar href="lib2.jar" />
</resources>

<application-desc main-class="org.wikipedia.es.JNLP" />

</jnlp>
```

5.4 Serialització[xviii]

En un context d'informàtica més modern, la serialització és el procés de salvar un objecte a un medi d'emmagatzematge (com pot ser un fitxer, o un buffer de memòria) amb la finalitat de transmetre'l a través d'una connexió en xarxa com una sèrie de bytes o en un format humanament més llegible com XML. La sèrie de bytes o el format es poden utilitzar per recrear un objecte que és idèntic en el seu estat intern a l'objecte original (de fet un clon). Aquest tipus de serialització s'utilitza principalment per transportar un objecte a través d'una xarxa, per persistir objectes a un fitxer o base de dades, o per a distribuir objectes idèntics a unes quantes aplicacions o localitzacions.

Aquest procés de serialitzar un objecte també s'anomena desinflar (deflating en anglès) un objecte o ordenar (marshalling en anglès) un objecte.

L'operació oposada, que extreu una estructura de dades d'una sèrie de bytes, és la deserialització (que s'anomena també inflating o unmarshalling).

5.4.1 Exemple d’utilització:

Per que un objecte sigui serialitzable, ha d’implementar l’ interfície: `java.io.Serializable` (que l’únic que fa es marcar l’objecte com serialitzable, sin que s’implementi cap mètode).

```
import java.io.Serializable;
public class Particula implements Serializable{
	private static final long serialVersionUID = 7114922815289308279L;
```

- Per que un objecte sigui serialitzable cal que totes les seves propietats siguin serialitzables.
- Tots els tipus primitius de Java son serialitzables per defecte (enters, cadenes, booleans etc...)
- La constant *`serialVersionUID`* serveix per identificar l’objecte quan l’estem llegint i poder assignar les dades al objecte corresponent.

Desar objectes

```
java.io.FileOutputStream f = new java.io.FileOutputStream(path);
java.io.ObjectOutputStream out = new java.io.ObjectOutputStream(f);
out.writeObject(obj);
out.flush();
out.close();
f.close();
```

Llegir Objectes

```
java.io.FileInputStream f =  new java.io.FileInputStream(path);
java.io.ObjectInputStream in = new java.io.ObjectInputStream(f);
Object o = in.readObject();
in.close();
f.close();
```

5.5 IDEs[14] utilitzats

Com a IDEs per al desenvolupament del projecte s'han utilitzat els següents programes:

Eclipse Galileo[xix] com la eina de programació principal, molt potent en la depuració de codi i per la "netedat" del codi, ja que no insereix res de codi addicional. Pero te mancances en la creació de vistes i pantalles.

Netbeans 6.9[xx] com la eina de programació secundaria. En les mancances gràfiques de l'anterior aquest ha estat l'encarregat de construir totes les vistes de l'aplicació.

Visual Paradigm[xxi] es la eina utilitzada per fer el modelatge de tot el UML d'aquest projecte: casos d'us, diagrames de classes etc.. A més es pot integrar amb els programes anteriors per facilitar els canvis que puguin produir en el procés d'implementació.

Subversion[xxii] es l'element indispensable en qualsevol projecte de desenvolupament de software, permet centralitzar un repositori de versions. Com a servidor se subversion s'ha utilitzat el que facilita la FIB per els estudiants[xxiii].

[14] Un IDE és un entorn de programació que ha estat empaquetat com un programa d'aplicació, és a dir, consisteix en un editor de codi, un compilador, un depurador i un constructor d'interfície gràfica (GUI). Els IDEs poden ser aplicacions per si soles o poden ser part d'aplicacions existents.

6 Planificació

La planificació temporal i econòmica del projecte implica l'organització de les diferents tasques i etapes d'aquest en el temps, per servir de guia des dels seus inicis fins al seu final.

Per claredat de l'organització de la memòria, aquesta planificació s'ha situat al final, però la seva realització va ser en acabar l'informe de definició del projecte, fent-se una segona iteració (que és la que es mostra) en acabar l'anàlisi de requisits i definir la metodologia. Aprofitant la seva ubicació en la part final de la memòria, podem revisar la planificació i veure quins són els desviaments ocorreguts i revisar si la previsió de riscos quant a l'esforç en hores era la correcta o no.

6.1 Definició de tasques

Tasca	**Hores aprox.**	**Risc estimat**	**Desviació hores**
1. Estudi previ de laboratoris web existents i aplicacions similars.	20	Baix	0
2. Investigar com exportar dades generades als experiments a un full de càlcul.	15	Baix	0
3. Investigar com realitzar les mesures de tots els experiments.	30	Baix	0
	55		0
4. Presa de requeriments (Reunions)	5	Baix	0
5. Anàlisi de requisits i estudi de viabilitat	20	Baix/Mig	+6
6. Analitzar utilització d’una Api per les mesures i l’exportació a Excel o fer la implementació pròpia.	10	Baix	-3
	38		+3
7. Especificació del model de dades.	20	Baix/MIg	+5

8. Disseny de l'estructura de programa.	10	Baix	+0
9.Definició del abast de projecte (+Reunions)	5	Baix	+0
	40		+5
10.Implementació iterativa (prototipat)	150	Mig/Alt	+45
11. Creació prototipus - proves	50	Baix/Mig	+10
	255		+55
12. Revisions i/o modificacions funcionals	20	Baix	+0
13. Inclusió de noves funcionalitats	10	Baix/Mig	+3
	33		+3
14. Documentació	75	Mig/Alt	+30
	105		+30
Total hores			**526**

El total d'hores del projecte són 430 hores, amb un desviament de 96 hores (22% de desviament d'hores). A continuació es mostra una taula de multiplicadors utilitzats per la desviació produïda a cada tasca en funció del risc que comporta.

Risc	**Multiplicador ± h**	**Risc**	**Multiplicador ± h**
Baix	0,1	Mig/Alt	0,7
Baix/Mig	0,3	Alt	1
Mig	0,5	Molt Alt	1,5

6.2 Assignació de tasques

L’assignació de tasques als perfils involucrats al projecte es un pas previ per poder fer la valoració econòmica del projecte.

Tasca	**Perfil**
1. Estudi previ de laboratoris web existents i aplicacions similars.	Analista
2. Investigar com exportar dades generades als experiments a un full de càlcul.	Analista
3. Investigar com realitzar les mesures de tots els experiments.	Analista
4. Presa de requeriments (Reunions)	Director de Projecte + Analista
5. Anàlisi de requisits i estudi de viabilitat	Analista
6. Analitzar utilització d’una Api per les mesures i l’exportació a Excel o fer la implementació pròpia.	Analista
7. Especificació del model de dades.	Analista
8. Disseny de l’estructura de programa.	Dissenyador
9.Definició del abast de projecte (+Reunions)	Director de projecte
10.Implementació iterativa (prototipatge)	Programador
11. Creació prototipus – proves	Programació
12. Revisions i/o modificacions funcionals	Analista
13. Inclusió de noves funcionalitats	Programador
14. Documentació	Analista (50%) + Programador (50%)

7 Valoració econòmica

Una vegada tenim definides tant les tasques com qui les realitzarà, podem fer la valoració econòmica del projecte. A mes del cost de les persones cal comptar el cost de software utilitzat més el hardware necessari per el desplegament del programa.

7.1 Cost de personal

Perfil persona	**Cost total/hora**	**Hores realitzades**	**Cost**
Director de projecte	40€	10	400€
Dissenyador	22€	10	220€
Analista Funcional	22€	185,5	4081€
Programador	18€	307,5	5535€
		Cost total	10.236€

7.2 Cost de software utilitzat

Per el desenvolupament del projecte s’han fet servir diferents recursos, aquests costos normalment es solen repercutir en els costos dels projectes. Degut a que s’ha utilitzat software lliure no hi ha cost a repercutir.

Recurs	**Ús**	**Cost**
Eclipse Galileo	Desenvolupament e implementació	0€
Netbeans 6.9	Desenvolupament e implementació	0€
Visual Paradigm UML	Especificació	0€
Subverion	Desenvolupament e implementació	0€

7.3 Cost de hardware per el desplegament

Per últim com a cost de projecte cal afegir els costos del desplegament de l'aplicació, que en aquest cas son els costos del servidor web que allotjarà la pàgina que té els enllaços per executar el laboratori web.

Descripció	Cost
Dell PowerEdge T110II com a servidor	436€
Microsoft Windows Server 2003 Llicencia de sistema operatiu	142,19€
APACHE HTTP SERVER Llicencia de servidor web	0€
VeriSign Certificat per la seguretat de la web + poder signar components del JNLP.	1299€/any

Per tant el cost total del projecte es de **12.113,19 €**. Dels quals 1299€ son costos anuals del servei certificat del lloc web.

8 Conclusions

En aquest punt del document com a resum de tot el que s’ha intentat explicar en aquest document, es pot dir el següent sobre els objectius inicials del projecte:

- S’ha implementat una eina capaç de fer simulacions completes, amb molta sortida d’informació que es pot explotar de moltes maneres. Per tant l’objectiu principal del projecte, sota el meu punt de vista, s’ha completat satisfactòriament.
- Es una eina que es pot fer créixer en un futur (per mi o altres persones) amb més experiments nous o amb millores funcionals o d’ usabilitat.
- Per la realització dels projecte s’han fet servir components externs, la decisió d’utilitzar aquests components ja implementats ha estat encertada ja que ha facilitat la implementació i ha donat molta més potència al programa.
- La idea principal del projecte era un laboratori a la web, però ja que el programa té l’estructura com una aplicació, pot ser utilitzat de manera independent sense requerir d’un lloc web.
- Crec que va ser una decisió encertada enfocar el projecte com una aplicació i no com un applet (que es el que els estudiants en projectes anteriors havien fet), ja que ha permès que sigui més polivalent, pot ser utilitzat en un lloc web o es pot distribuir com un programa propi.
- Com a comentari sobre productes similars, m’ha semblat curiós que no existeixi cap producte similar a cap universitat espanyola (o no l’he trobat). Crec que podria potenciar-se més.

Comprant el meu projecte amb el laboratoris analitzats i altres productes, es pot concloure que en aspectes gràfics i en numero de experiments són molt millors, però en canvi en extracció i explotació de dades, el projecte es superior. Es podria dir que té més possibilitats didàctiques, per exemple, es podria utilitzar en sessions pràctiques on l’estudiant ha d’extreure conclusions a partir de probes i resultats numèrics.

Degut a la meva disponibilitat personal per a la realització del projecte s’ha allargat més d’una any, quant per planificació podria haver estat enllestit en 5-6 mesos.

Gracies a la realització del projecte he pogut tornar a refrescar tot el que havia après de física, també m’ha servit per recordar com m’agrada llegir coses sobre el món de la física, els treballs dels grans físics i matemàtics de la historia.

9 Propostes de millora

Encara que s'han aconseguit els objectius i els requisits marcats a l'inici del projecte s'han cobert, sempre sorgeixen noves idees i línies de continuació, com es detallen a continuació:

Millores funcionals:

- **Generació de gràfics en viu:** Possibilitat de veure la representació de dades en directe, mentre s’executa el programa.
- **Regulació de la velocitat dels experiments:** Poder regular la velocitat dels experiments per poder obtenir més detall o poder veure més de pressa.
- **Interacció amb el panell del experiment:** Permetria modificar les propietats dels elements del experiment, per exemple: si tenim una partícula al experiment, poder canviar la posició, aplicar força etc.. Tots això s’hauria de fer mitjançant propietats del panell i dels esdeveniments del ratolí.
- **Afegir càlcul d’energia i treball:** Per entendre millor el que està succeint al experiment seria interessant afegir barres o nivells de energies, per exemple, una barra amb la quantitat de energia potencial i energia cinètica en un tir parabòlic, on es podria observar els canvis que es produeixen.
- **Experiments en 3D:** Per fer experiments més potents es podrien generar en 3D, aprofitant la potencia de Java per el desenvolupament.

Millores d' usabilitat:

- **Zoom al panell del experiment:** Incloure una opció per poder regular la grandària d'aquest i per tant poder fer zoom
- **Obrir el fulls de càlcul:** Una vegada exportades les dades a un full de càlcul, la possibilitat d'obrir el programa editor (si esta instal·lat a la màquina).
- **Resolució variable:** La grandària de l'aplicació es podria regular en funció de la resolució del ordinador on s'executi el programa

10 Agraïments

Com a punt final voldria agrair a tota la gent que m'ha ajudat a fer aquest projecte: en Jordi Martí, per donar-me suport en totes les meves mancances en física, també per donar-me bons consells i guiar-me durant tot el projecte.

Voldria donar les gracies a Angel Franco García (Doctor en Ciències Físiques – EHU Euskal Herrico Unibertsitatea), per la seva web[xxiv] que m'ha servit de referència i per resoldre dubtes.

Els meus companys d'universitat: l' Isaac, el Jordi, el Raúl, la Míriam per ajudar-me en tot aquest camí que culmina amb aquest projecte. Amb ells he compartit molt bons moments en treballs, practiques etc..

A la meva família i amics per entendrem i suportar-me durant la realització del projecte, per motivar-me i donar ànims.

També agrair als companys de feina per entendre la meva situació i deixar-me poder acabar-ho.

Gracies a tots ells, sense ells això no hagués estat possible.

11 Bibliografia

- Patterns of enterprise application architecture, Martin Fowler et al., Addison-Wesley, cop. 2003, ISBN 0321127420
- Java Swing, Marc Loy - Robert Eckstein - Dave Wood - James Elliot and Brian Cole , cop. 2003, ISBN 0596004087
- Física para la ciencia y la tecnología: Electricidad y magnetismo, luz. Tipler- Mosca, cop 2003, ISBN 9788429144123
- Física Giancoli – Giancoli C.Douglas cop.2006 ISBN 9762607760

12 Annexos

Mètodes de Runge-Kutta de segon ordre

La versió de segon ordre de l'equació $y_{i+1} = y_i + \varphi(x_i, y_i, h)h$ és:

$y_{i+1} = y_i + (a_1 k_1 + a_2 k_2)h$

On:

$k_1 = f(x_i, y_i)$

$k_2 = f(x_i + p_1 h, y_i + q_{11} k_1 h)$

Els valors de a_1, a_2, p_1 i q_{11} són avaluats en igualar el terme de segon ordre de l'equació donada amb l'expansió de la sèrie de Taylor. Es desenvolupen tres equacions per avaluar quatre constants desconegudes. Les tres equacions són:

$$a_1 + a_2 = 1 \qquad a_2 p_1 = \frac{1}{2} \qquad a_2 q_{11} = \frac{1}{2}$$

Com es tenen tres equacions amb quatre incògnites s'ha de suposar el valor d'una d'elles. Suposant que es va especificar un valor para a_2, es pot resoldre de manera simultània el sistema d'equacions obtingut:

$$a_1 = 1 - a_2 \qquad p_1 = q_{11} = \frac{1}{2a_2}$$

Com es pot triar un nombre infinit de valors para $\boldsymbol{a_2}$, hi ha un nombre infinit de mètodes Runge-Kutta de segon ordre. Cada versió podria donar exactament els mateixos resultats si la solució de l'EDO fos quadràtica, lineal o una constant.

$a_2 = 1/2$: Mètode de Heun amb un sol corrector, on:

$$y_{i+1} = y_i + \left(\frac{k_1}{2} + \frac{k_2}{2}\right)h \qquad On: \begin{cases} k_1 = f(x_i, y_i) \\ k_2 = f(x_i + h, y_i + k_1 h) \end{cases}$$

$a_2 = 1$: Mètode del punt mig, on:

$$y_{i+1} = y_i + k_2 h \quad On: \begin{cases} k_1 = f(x_i, y_i) \\ k_2 = f\left(x_i + \frac{h}{2}, y_i + k_1 \frac{h}{2}\right) \end{cases}$$

$a_2 = 2/3$: Mètode de Ralston:

$$y_{i+1} = y_i + \left(\frac{k_1}{3} + \frac{2k_2}{3}\right)h \qquad On: \begin{cases} k_1 = f(x_i, y_i) \\ k_2 = f\left(x_i + \frac{3h}{4}, y_i + 3k_1\frac{h}{4}\right) \end{cases}$$

Mètodes de Runge-Kutta de tercer ordre

Per n = 3 el resultat són sis equacions amb vuit incògnites, per tant s'han de suposar dos valors amb antelació per poder desenvolupar el sistema d'equacions. Una versió àmpliament usada és:

$$y_{i+1} = y_i + \left(\frac{k_1 + 4k_2 + k_3}{6}\right)h \quad On: \begin{cases} k_1 = f(x_i, y_i) \\ k_2 = f\left(x_i + \frac{h}{2}, y_i + k_1\frac{h}{2}\right) \\ k_3 = f(x_i + h\,, y_i - k_1 h + 2k_2 h) \end{cases}$$

Mètodes de Runge-Kutta de quart ordre

Un membre de la família dels mètodes Runge-Kutta és usat tan comunament que sovint és referenciat com "RK4" o com "el mètode Runge-Kutta".

Definim un problema de valor inicial com:

$$y' = f(x, y), \quad y(t_0) = y_0$$

Llavors el mètode RK4 per a aquest problema està donat per la següent equació:

$$y_{i+1} = y_i + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4\,)h$$

$$k_1 = f(x_i, y_i)$$

$$k_2 = f\left(x_i + \frac{1}{2}\,h, y_i + \frac{1}{2}k_1 h\right)$$

$$k_3 = f\left(x_i + \frac{1}{2}\,h, y_i + \frac{1}{2}k_2 h\right)$$

$$k_4 = f(x_i + h, y_i + k_3 h)$$

Així, el següent valor (y_{n+1}) és determinat pel present valor (y_n) mes el producte de la grandària de l'interval h per un pendent estimat. El pendent és una mitjana ponderada de pendents:

k_1 és el pendent al principi de l'interval.

k_2 és el pendent en el punt mitjà de l'interval, usant k_1 per determinar el valor d'y en el punt $t_n + \frac{h}{2}$ usant el mètode de Euler.

k_3 és una altra vegada el pendent del punt mitjà, però ara usant k_2 per determinar el valor d'y.

k_4 és el pendent al final de l'interval, amb el valor d'y determinat per k_3

Fent la mitjana dels quatre pendents, se li assigna major pes als pendents en el punt mitjà:

$$pendent = \frac{k_1 + 2k_2 + 2k_3 + k_4}{6}$$

Aquesta forma del mètode de Runge-Kutta, és un mètode de quart ordre la qual cosa significa que l'error per pas és de l'ordre $d'O(h^5)$, mentre que l'error total acumulat té l'ordre $O(h^4)$.

13 Referències

[i] Web del departament de Física i Enginyeria Nuclear:
http://dfen.upc.edu/docencia/projectes-docents/laboratori-virtual-de-fisica

[ii] Mètodes Numèrics:
http://www.salle.url.edu/~oscarg/resources/dinam0/dpja1.html

[iii] Mètode d'Euler:
http://ca.wikipedia.org/wiki/M%C3%A8tode_d%27Euler

[iv] Mètode Runge-Kutta:
http://es.wikipedia.org/wiki/M%C3%A9todo_de_Runge-Kutta

[v] Llei d'Stokes:
http://es.wikipedia.org/wiki/Ley_de_Stokes

[vi] Michael Faraday
http://ca.wikipedia.org/wiki/Michael_Faraday

[vii] Newton(unitat)
http://ca.wikipedia.org/wiki/Newton_%28unitat%29

[viii] Coulomb(unitat)
http://ca.wikipedia.org/wiki/Coulomb

[ix] Volt(unitat))
http://ca.wikipedia.org/wiki/Volt

[x] Llei de Coulomb:
http://ca.wikipedia.org/wiki/Llei_de_Coulomb

[xi] Tesla:
http://ca.wikipedia.org/wiki/Tesla_%28unitat%29

[xii] Força de Lorentz:
http://ca.wikipedia.org/wiki/For%C3%A7a_de_Lorentz

[xiii] Equacions de Maxwell:
http://ca.wikipedia.org/wiki/Equacions_de_Maxwell

[xiv] Principi de conservació del moviment lineal:
http://www.sc.ehu.es/sbweb/fisica/dinamica/con_mlineal/dinamica/dinamica.htm#Conservaci%C3%B3n%20del%20momento%20lineal%20de%20un%20sistema%20de%20part%C3%ADculas

[xv] JScience
http://jscience.org/

[xvi] Java Excel Api:
http://jexcelapi.sourceforge.net/

[xvii] JNLP:
http://download.oracle.com/javase/1.5.0/docs/guide/javaws/developersguide/syntax.html

[xviii] Java Serialization
http://download.oracle.com/javase/6/docs/technotes/guides/serialization/

[xix] Eclipse Galileo:
http://www.eclipse.org

[xx] Netbeans 6.9
http://netbeans.org/

[xxi] Visual Paradigm
http://www.visual-paradigm.com/

[xxii] Subversion
http://subversion.tigris.org/

[xxiii] Página personal Subversion
https://svn.fib.upc.es/svn/david.perez-alberto/PFC_APPLET_F/

[xxiv] Física con ordenador
http://www.sc.ehu.es/sbweb/fisica/default.htm