

TRNAVSKÁ UNIVERZITA V TRNAVE

Pedagogická fakulta

Fakulta fyziky

Diplomová práca

IP-COACH

Ján Salva

2013

IP-COACH

Diplomová práca

Ján Salva

TRNAVSKÁ UNIVERZITA V TRNAVE

Pedagogická fakulta

Katedra fyziky

Učiteľstvo akademických predmetov

v kombinácii predmetov

fyzika – informatika

Vedúci práce: doc. RNDr. Peter Čerňanský, PhD.

Trnava

2013

ABSTRAKT

Témou práce je analyzovať problematiku voľného pádu vo vákuu a s odporom prostredia. Opísať systémy ktoré sa používajú pri tvorbe e-experimentov a fyzikálnych modelov pri výučbe fyziky na stredných a vysokých školách. Práca je rozdelená do 6. kapitol. Prvá nás uvedie do problematiky spracovania experimentov pomocou dostupných systémov. V druhej a tretej kapitole je spracovaný experiment voľného pádu v systéme ISES. Vo štvrtej kapitole je popísaná problematika simulácie fyzikálnych dejov a spracovanie simulácie voľného pádu v programe EJS. Posledné dve kapitoly patria záveru a prílohám. Cieľom práce je oboznámiť čitateľa o možnostiach spracovania experimentov na vyučovanie fyziky.

Kľúčové slová: voľný pád, experiment, odpor prostredia, EJS, ISES,

ABSTRAKT

This is work.

Zoznam ilustrácií

Obrázok 1: Vyhladenie krivky metódou Brazier.....	15
Obrázok 2: Vyhladenie grafu metódou Spline s hodnotou 0 a hodnotou 1.....	16
Obrázok 3: Filtrovanie krivky grafu.....	17
Obrázok 4: derivácia grafu pred filtrovaním a po filtrovaní.....	17
Obrázok 5: Programovacie prostredie Robolab.....	20
Obrázok 6: Jednoduchý Pilot program.....	21
Obrázok 7: Okno programu ISES.....	24
Obrázok 8: Závislosť dráhy od času.....	29
Obrázok 9: Závislosť rýchlosti od času.....	30
Obrázok 10: Závislosť zrýchlenia od času.....	30
Obrázok 11: Hlavné okno programu EJS.....	41
Obrázok 12: Panel Description EJS.....	43
Obrázok 13: View panel EJS.....	45
Obrázok 14: Zadávanie vzorcov v EJS.....	47
Obrázok 15: Definovanie premenných v EJS.....	47
Obrázok 16: Okno spustenej simulácie EJS.....	48

Predhovor

Pre spracovanie tejto témy som sa rozhodol, lebo myšlienka e-experimentov a simulovaný fyzikálnych dejov sa mi zdá zaujímavá. Simulovanie fyzikálnych dejov sa často vyžíva vo vyučovacom procese fyziky a poznatky získané pri experimentoch sú dlhodobejšie. V dnešnej dobe kedy je záujem o fyziku na ústupe a o predmet ako taký nie je záujem je toto jedna z možných ciest ako propagovať fyziku v školskom prostredí a pritiahnuť pozornosť mladých ľudí. Počítače sa stali našou bežnou súčasťou života a zostavenie a prevádzka takýchto e-experimentov nieje náročná. Aj preto verím, že tento spôsob výučby má význam a potenciál do budúcnosti a je tou správnou cestou ktorou sa má vyučovanie fyziky uberať.

V diplomovej práci som popísal základné systémy na vytváranie e-experimentov. Súčasťou práce je rozobrať voľný pád na teoretickej úrovni a meranie voľného pádu vo vákuu a s odporom prostredia v systéme ISES. Taktiež som sa zameral na simuláciu fyzikálnych dejov v prostredí Easy Java Simulation. Je to jednoduché prostredie ktoré je vhodné na simuláciu rôznych dejov pre učiteľov fyziky, ale aj iných prírodovedných predmetov. Umožňuje priniesť do výučby fyziky dlho chýbajúci prvok a to je názornosť. Vďaka tomuto systému môžu školy vnieť do hodín fyziky experimenty bez nutnosti budovania finančne nákladných laboratórií. Tento modelačný systém je šírený pod licenciu opensource. Čo ponúka ďalšie možnosti úprav a prispôsobenia. Vo svete je Easy Java Simulation rozšírený a má širokú podporu. Na internete je dostupné veľké množstvo už nasimulovaných fyzikálnych dejov ktoré sú dostupné zadarmo.

V práci som vypracoval simuláciu voľného pádu ktorá sa nachádza v prílohách. A jej popis je uvedený v štvrtej kapitole.

ČESTNÉ PREHLÁSENIE

Prehlasujem, že som zadanú diplomovú prácu vypracoval samostatne, pod odborným vedením vedúceho diplomovej práce doc. RNDr. Peter Čerňanský, PhD. a použil som len literatúru uvedenú v práci.

V Trnave, dňa 18.1.2013

Ján Salva

Obsah

Úvod.....	12
1 Počítačom podporované experimenty.....	13
1.1 Využitie počítača ako nástroja na meranie.....	13
1.2 Prostredie pre meranie na počítačoch.....	14
1.3 IP-COACH.....	14
1.3.1 Základné funkcie IP-COACH.....	15
1.4 RoboLab.....	20
1.4.1 Programovanie.....	20
1.4.2 Pracovné prostredie.....	21
1.5 ISES.....	26
1.5.1 Prostredie programu.....	27
1.5.2 Meranie a organizácia dát.....	29
1.5.3 Zobrazenie dát.....	30
2. Teoretický popis voľného pádu vo vákuu.....	30
2.1 Popis voľného pádu.....	31
2.2 Grafické znázornenie voľného pádu.....	31
2.3 Matematický popis voľného pádu.....	35
2.3.1 Rovnice voľného pádu vo vákuu.....	35
3 Teoretický popis voľného pádu s odporom prostredia.....	36
3.1 Matematický popis.....	37
3.2 Rovnice voľného pádu so zohľadnením odporu prostredia.....	37
3.3 Odvodenie vzorcov.....	38
4 Voľný pád v systéme ISES.....	39
4.1 Popis experimentu.....	40
4.2 Princíp experimentu.....	40
5 Simulácia voľného pádu v prostredí EJS.....	43
5.1 Popis prostredia EJS.....	43
5.2 Popis panelov EJS.....	46
5.2.1 Description panel.....	47
5.2.2 Model panel.....	48
5.2.3 View panel.....	49
5.3 Popis simulácie.....	51

Záver.....	54
Bibliografia.....	55

Úvod

Využívanie moderných technológií je dnes už neoddeliteľnou súčasťou nášho života. Rozvojom elektroniky a výpočtovej techniky sa nám otvárajú nové možnosti v spoznávaní a odhaľovaní tajomstiev okolitého sveta. Tento potenciál sa dá využiť v rôznych sférach poznávania a bola by obrovská škoda ho nevyužiť aj v školách na vyučovaní fyziky. Počítače dokážu nahradiť veľké množstvo meracej techniky čím sa môže doceliť nemalá úspora finančných prostriedkov. V slovenských školách sa začali presadzovať rôzne meracie systémy, ktoré ponúkajú názorný výklad preberanej témy na hodinách fyziky. Najrozšírenejšie sú systém ISES, RoboLab a IP-COACH.

V prvej časti diplomovej práce som sa zamerail hlavne na popis troch najpoužívanejších systémov používaných na slovenských školách. Snažil som sa o stručné zhrnutie možností ktoré tieto systémy ponúkajú.

V druhej a tretej časti diplomovej práce som sa snažil zostaviť experiment meraný pomocou systému ISES. Experimentálne dokázať odpor prostredia pri voľnom páde. Rozobrať problematiku voľného pádu vo vákuu a voľného pádu s odporom prostredia. Matematicky popísať voľný pád. Získané hodnoty som overil teoreticky.

Vo štvrtej časti práce som sa zamerail na simuláciu voľného pádu pomocou simulačného softvéru Esay Java Simulation. Vypracoval som simuláciu ktorá umožňuje názorne skúmať charakter voľného pádu. Táto simulácia sa dá využiť vo vyučovaní fyziky ako názorný experiment. Je súčasťou práce a nachádza sa v prílohách.

1 Počítačom podporované experimenty

V dnešnej dobe označovanej ako počítačová doba nájdeme počítače vo všetkých oblastiach ľudskej činnosti. Počítače zaujali svoje miesto samozrejme aj v školách. Veľký potenciál majú v oblasti fyziky, kde sa dajú okrem štandardného využitia pri prezentovaní a matematických výpočtoch veľmi dobre využiť ako meracie prístroje. Vďaka rôznym druhom softvéru a AD/DA prevodníkom môžeme pomocou počítača simulovať veľké množstvo meracích prístrojov.

1.1 Využitie počítača ako nástroja na meranie

Počítač bez dodatočných zariadení nieje schopný zmerať analógové veličiny nakoľko pracuje s digitálnym nespojitým signálom. Pomocou analógovo-digitálneho prevodníka môžeme spojiť signáli previesť na signál digitálny, ktorý vie počítač spracovať a následne pomocou vhodného softvéru vyhodnotiť.

Na meranie ostatných fyzikálnych veličín sa používa obdobný spôsob. Meranie veličiny neprebieha priamo, ale sa pomocou vhodného senzora prevádza na elektrické napätie ktoré dokáže počítač spracovať a vyhodnotiť. Takýmto spôsobom môžeme merať aj neelektrické veličiny ako sú napríklad tlak, žiarenie, teplotu.

Používanie počítača na hodinách fyziky má aj iné výhody. Nenahradzuje len samotné meracie prístroje, ale ponúka nám oveľa väčšiu paletu možností. Pomocou softvéru môžeme získané výsledky z meraní spracovávať, tvoriť grafy, prevádzať s nimi zložité matematické operácie. Počítače nám tiež ponúkajú možnosť simulácii dejov reálneho prostredia. Simulácia môže prebiehať ako vzdialený experiment, alebo naprogramovaný dej.

1.2 Prostredie pre meranie na počítačoch

Dnes existuje veľké množstvo systémov komplexných riešení, ktoré sú ponúkané školám ako sady prevodníkov, modulov a softvéru k účelu merania na počítači. V každej krajine má väčšinový podiel trhu jedno alebo dva riešenia. Vo svete je dosť rozšírený systém LOGOTRON, je to robustný systém ktorý nám umožňuje grafické zobrazenie závislostí.

Na Slovensku sú to hlavne systémy IP-COACH a ISES v menšej miere ešte systém RoboLab.

1.3 IP-COACH

Meracie rozhranie IP-COACH je univerzálny nástroj na vzdelávanie v rôznych smeroch ako sú prírodovedné vedy, matematika, fyzika a podobne. Obsahuje sadu nástrojov vhodných na meranie, riadenie, videomeranie, modelovanie, analyzovanie dát a podobne. Pri meraní zbiera rozhranie IP-COACH dáta zo senzorov ktoré sú pripojené. Na výber máme veľké množstvo senzorov ako sú senzor teploty, pH, osvetlenia, uhlového zrýchlenia, pootočenia a podobne. Riadenie IP-COACH nám umožňuje programovať rôzne programy pomocou ktorých môžeme ovládať silové prvky ako sú žiarovky, motor, relé. Takýmto spôsobom sa dá naprogramovať výkonné automatizované meranie. Programovacie prostredie nám ponúka viac druhov rozhraní od jednoduchého až po komplexné kedy máme na výber veľké množstvo riadiacich príkazov a funkcií. Prostredie IP-COACH nám tiež ponúka možnosť modelovania kde môžeme vytvárať dynamické modely v textovom, alebo grafickom móde. Dáta ktoré získame pomocou modelovaného deja je možné porovnať s dátami ktoré sme reálne namerali. Pomocou videomerania môžeme zbierať dáta zo sekvencie obrázkov, alebo videozáznamu.

Všetky dáta ktoré získame pri riadení, videomeraní, modelovaní, môžeme ďalej spracovávať nástrojmi ktoré nám ponúka rozhranie IP-COACH a sú súčasťou dodávaného systému. IP-COACH nám taktiež ponúka prostredie na úpravu, alebo tvorbu vlastnej aktivity.

Štruktúra IP-COACH merania predstavuje jednotlivé aktivity zoskupené do projektu. Projekt v IP-COACH predstavuje sadu aktivít ktorých vzájomné prepojenie môže, ale nemusí mať súvislosť. Každý projekt obsahuje krátke zhrnutie a návod vo forme textu a obrázka.

Zobrazenie dát prebieha pomocou rozhrania IP-COACH, kde dáta môžeme zobrazovať ako merací prístroj - zobrazí aktuálnu hodnotu senzora v grafickej škále. Hodnota – zobrazí hodnotu senzora, analógového vstupu, výsledok vzorca v digitálnej forme. Ďalej môžeme dáta zobrazit' ako graf, alebo tabuľku. Tabuľka môže obsahovať najviac šesť stĺpcov. Stĺpcom môžeme priradiť rôzne hodnoty, čas, hodnoty získané senzormi, manuálne zadané hodnoty, hodnoty vypočítané vzorcami. Pri výpočtoch zo vzorcov, musíme do vzorca zadávať len hodnoty ktoré už boli použité v predchádzajúcich stĺpcoch. Z takto zostavenej tabuľky môžeme zobrazit' ľubovoľný graf. Taktiež môžeme dáta triediť podľa ľubovoľného vzorca vzostupne, alebo zostupne.

1.3.1 Základné funkcie IP-COACH

Kalibrácia senzorov

Sériovo vyrábané senzory zvyčajne nemusíme kalibrovať. V IP-COACH nájdeme knižnicu senzorov CMA, ale aj množstvo kompatibilných senzorov od iných výrobcov. Samozrejme kvôli presnosti merania je vhodné kalibráciu pravidelne kontrolovať aj pri týchto senzoroch. IP-COACH nám ponúka možnosť vytvoriť si vlastné senzory a pomocou kalibrácie ho nastaviť na správne meranie veličiny. Súčasťou systému je kvalitný nástroj na kalibrovanie senzorov.

Nastavenie a spúšťanie merania

Pri nastavení merania je dôležitým údajom vzorkovacia frekvencia merania. Tá nám udáva počet zaznamenaných bodov počas merania. IP-COACH má možnosť nastaviť meranie manuálne, kedy sa zaznamenávajú merané hodnoty pri kliknutí myši, alebo stlačením klávesnice enter. Spustiť meranie môžeme aj pomocou úrovne meraného signálu z pripojených senzorov.

Spracovanie a analýza dát

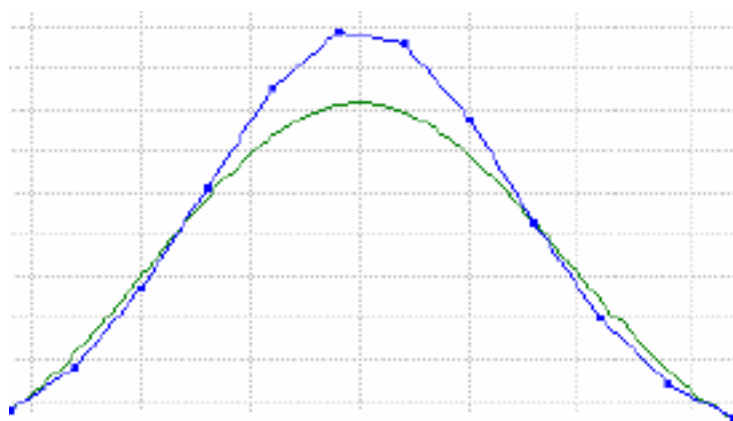
Pomocou dodávaných nástrojov spolu s IP-COACH môžeme upravovať a spracovávať dáta. Dáta môžeme zobrazit' v grafoch, tabuľkách, text. Jedným z najlepších spôsobov prezentovania dát je graf. Aj tu nám IP-COACH ponúka širokú škálu dostupných nástrojov.

Výber odstránenie dát v grafe pomocou funkcie Process->Select->remove data v ponuke nástrojov, môžeme vybrať interval dát, alebo jednotlivé body grafu. A následne ich odstrániť, alebo uchovať. Dáta ktoré odstránime sa zmažú v každom grafe a tabuľke aktívnej aktivity.

Vyhľadenie krivky

Pomocou funkcie Proces->Smoth graph v nástrojoch, môžeme aproximovať krivku grafu. Môžeme si vybrať z dvoch možností aproximácie Bezier a Spline. Vyhľadením grafu sa znásobí počet bodov v grafe, takto upravený graf môžeme ďalej spracovávať napríklad derivovať. Vyhľadenie grafu sa používa hlavne na potlačenie šumov a náhodných chýb merania.

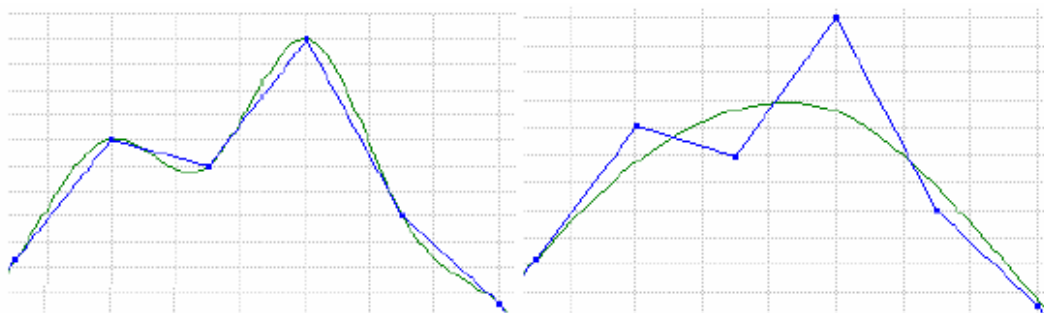
Metóda Bezier – vyhladený graf obsahuje prvý a posledný z pôvodných bodov. Tvar vypočítaného grafu závisí od všetkých bodov. Krivka smeruje k pôvodnej krivke grafu, no nemusí obsahovať pôvodné body. Počet pôvodných bodov, nameraných hodnôt musí byť menší ako 100. Znížiť počet nameraných bodov je možné operáciou Select->Remove data. Po vyhladení krivky môže byť počet bodov v závislosti na nastavení systému až 16 380.



Obrázok 1: Vyhľadenie krivky metódou Brazier

Aproximác

ia Spline – Metóda aproximácie Spline vyhladí krivku pomocou najmenších štvorcov. Tvar krivky v určitej oblasti závisí od polohy susedných bodov. Miera aproximácie je daná parametrom intervalu. Pri nastavení hodnoty intervalu rovnej 0 krivka grafu prechádza všetkými pôvodnými bodmi. Pri nastavení vyššej hodnoty intervalu sa bude graf blížiť k priamke.



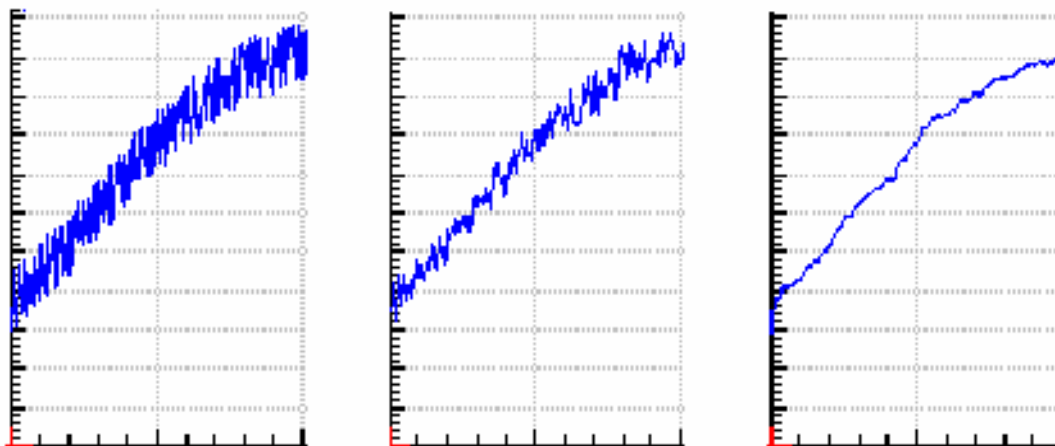
Obrázok 2: Vyhladenie grafu metódou Spline s hodnotou 0 a hodnotou 1

Vyhladený graf môžeme pridať do tabuľky grafu funkciou Breizer, alebo taktiež funkciou Spline. V takomto prípade bude nový vyhladený graf obsahovať rovnaký počet nameraných hodnôt, bodov ako pôvodný graf.

Odstránenie šumu z grafu, filtrovanie

Pri vzniku náhodných chýb a šumov pri meraní sa používa funkcia Filter. Výsledkom použitia funkcie je vyhladený graf ktorý obsahuje rovnaký počet nameraných hodnôt, bodov. Funkcia nahradzuje každý bod grafu priemernou hodnotou intervalu susedných bodov, čím sa dosiahne vyhladenie grafu. Parameter interval, určuje koľko bodov sa má pri výpočte použiť. Pri vhodne zvolenom intervale je každý bod nahradený priemernou hodnotou n predchádzajúcich, n nasledujúcich bodov a priemernou hodnotou samého seba. Funkciu filtrovania grafu sa odporúča použiť pred výpočtom derivácie funkcie. Hodnota vhodne zvoleného parametra intervalu závisí od počtu nameraných bodov. Použitím veľkej a nevhodne zvolenej hodnoty intervalu, môžu zaniknúť dôležité údaje grafu ako sú úzke minimá a maximá. Naopak pri zvolení malej hodnoty intervalu sa môže stať, že odfiltrujeme málo šumu a graf nebude dostatočne vyhladený. Vo všeobecnosti je lepšie filtrovať s malou hodnotou intervalu a viac krát. Jednotlivé body ležiace mimo krivky grafu

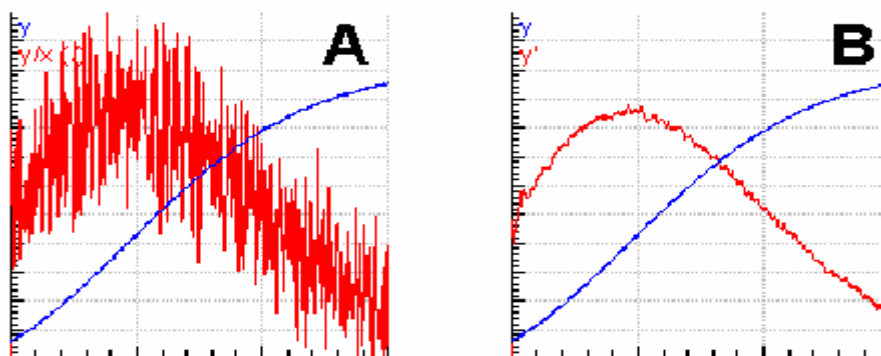
danej okolitými bodmi, je dobré pred použitím funkcie filtrovania odstrániť. Ak takéto body neodstránime a nevhodne zvolíme interval filtrovania, toto môže viesť k veľkým nepresnostiam merania.



Obrázok 3: Filtrovanie krivky grafu

Derivácia

Programové prostredie IP-COACH umožňuje derivovať funkciu. Pri derivácii sa počíta rozdiel dvoch susedných hodnôt zo stĺpca C2 a delí sa rozdielom hodnôt zo stĺpca C1. Pri malom počte nameraných bodov môže dochádzať k skresleniu výpočtu. Malé a náhodné výchylky v grafe funkcie výrazne ovplyvňujú deriváciu. Pred derivovaním je vhodné graf filtrovať, alebo upraviť pomocou funkcie vyhladenie grafu.



Obrázok 4: derivácia grafu pred filtrovaním a po filtrovaní

Sklon grafu

Táto funkcia slúži na manuálne zistenie sklonu dotýčnice ku grafu v ľubovoľnom bode. Pri použití tejto funkcie sa graf nastaví do režimu skenovania. Po kliknutí na ľubovoľný bod grafu v ktorom chceme zistiť sklon, sa objaví priamka ktorou môžeme otáčať myšou pri stlačení klávesy CTRL. Priamku do smeru dotýčnice musíme umiestniť manuálne. IP-COACH neukladá hodnoty sklonu dotýčnice, preto si ich musíme zapisovať napríklad do poznámkového bloku.

Plocha pod grafom

Prostredie IP-COACH nám tiež ponúka funkciu integrovania. Táto funkcia sa využíva na určenie plochy medzi grafom a súradnicou x , v intervale hodnôt súradnice x . Na zadávanie intervalu súradnice x nám slúži jednoduché grafické okno. Po zadaní intervalu nám program vypočíta plochu pod krivkou.

Fit funkcie

Táto funkcia nám ponúka aproximáciu dát grafu štandardnou matematickou funkciou. Koeficient fitovanej funkcie sa určuje metódou najmenších štvorcov. Fitovať môžeme automaticky, alebo manuálne, ale taktiež program ponúka aj kombináciu oboch predchádzajúcich možností. Tu môžeme začať fitovanie manuálne a potom ponechať program spresniť koeficienty automaticky.

1.4 RoboLab

Robolab je komplexný systém dodávaný do škôl. Balík obsahuje softvér pre programovanie a riadenie RCX. Programovacie prostredie je grafické prispôsobené pre deti. Samotné programovanie je realizované pomocou vývojových diagramov čím sa stáva lákavým a prítiažlivým prostredím pre deti. Programovanie je intuitívne.

Robolab vyvinul Dr. Chris Rogers so svojim vývojovým tímom na Tuftovej univerzite, Massachusetts USA. Za základ zobral LabVIEW, softvér určený pre vývoj aplikácií pre simuláciu, modelovanie, riadenie a výskum. Robolab teda využíva modularitu a výhody tohto systému, narozdiel od neho je, ale prispôsobený pre deti.

Práca v Robolabe

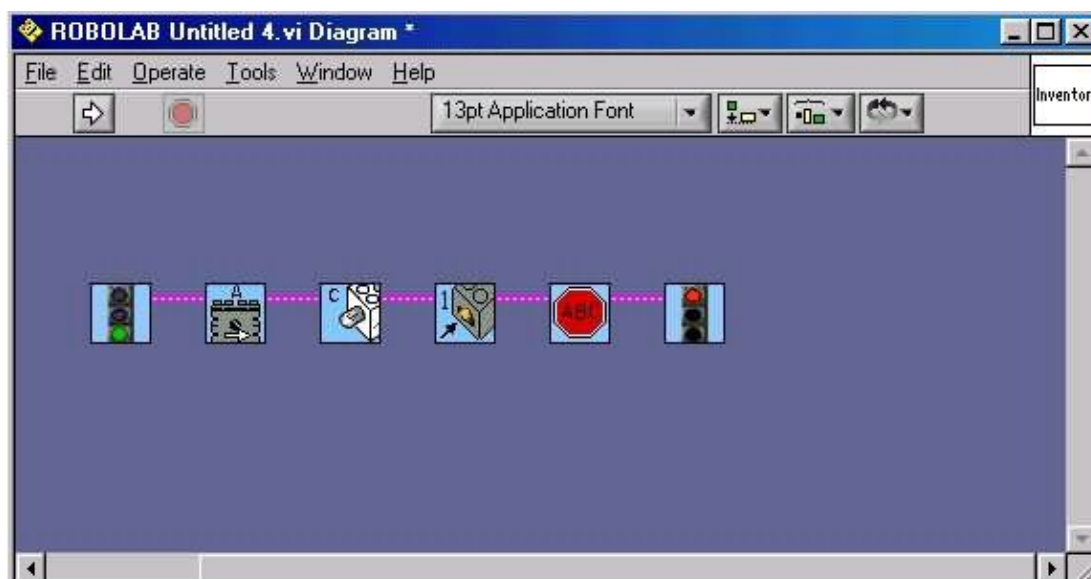
Pracovať môžeme v troch režimoch:

1. Administrátor slúži pre správu a údržbu programu a RCX. V tomto režime môžeme nahrávať firmware a podobne.
2. Programmer slúži pre písanie programov. K dispozícii máme dve úrovne programovania pilot a inventor.
3. Najužívanejší režim je investigator. Môžeme v ňom vytvoriť celý vedecký projekt. Navrhnuť textové zadania, navrhnuť program, pripojiť fotografiu, alebo návod na meranie, videoklip, vložiť prehľadnú tabuľku nameraných hodnôt.

1.4.1 Programovanie

Programovanie v Robolabe

K programovaniu Robolabu je navrhnutý plne grafický ikonický programovací jazyk. Zdrojový kód programu teda predstavuje vývojový diagram. Ikony jednotlivých príkazov sú spolu pospájané akýmisi spojmi, ktoré určujú po ktorej linke má program bežať. Príklad takéhoto programovania je vidieť na obrázku.



Obrázok 5: Programovacie prostredie Robolab

Dieťa teda programuje tak, že do okna programu umiestňuje ikony reprezentujúce jednotlivé príkazy, a tie potom spája spojkami podľa smeru postupu algoritmu programu. Prostredie je dostatočne intuitívne sofistikované, takže samo ponúkne správne spájanie príkazov, ak ikony umiestňujeme dostatočne blízko seba. Samozrejme nám prostredie ponúka aj možnosť príkazy programu pospájať ručne, takmer ľubovoľným spôsobom. Pri práci v programovacom prostredí nám pomáha plávajúca nápoveda, ktorá je jednou z najväčších predností programu. Ku každému príkazu je podrobný popis spolu s ukázkovým programom. Ten môžeme rovno nahrávať do jednotky RCX a sledovať výslednú činnosť programu. Jednotlivé príkazy sa vyberajú z palety príkazov ktorá sa dá modifikovať a môže nám ponúknuť len príkazy ktoré potrebujeme.

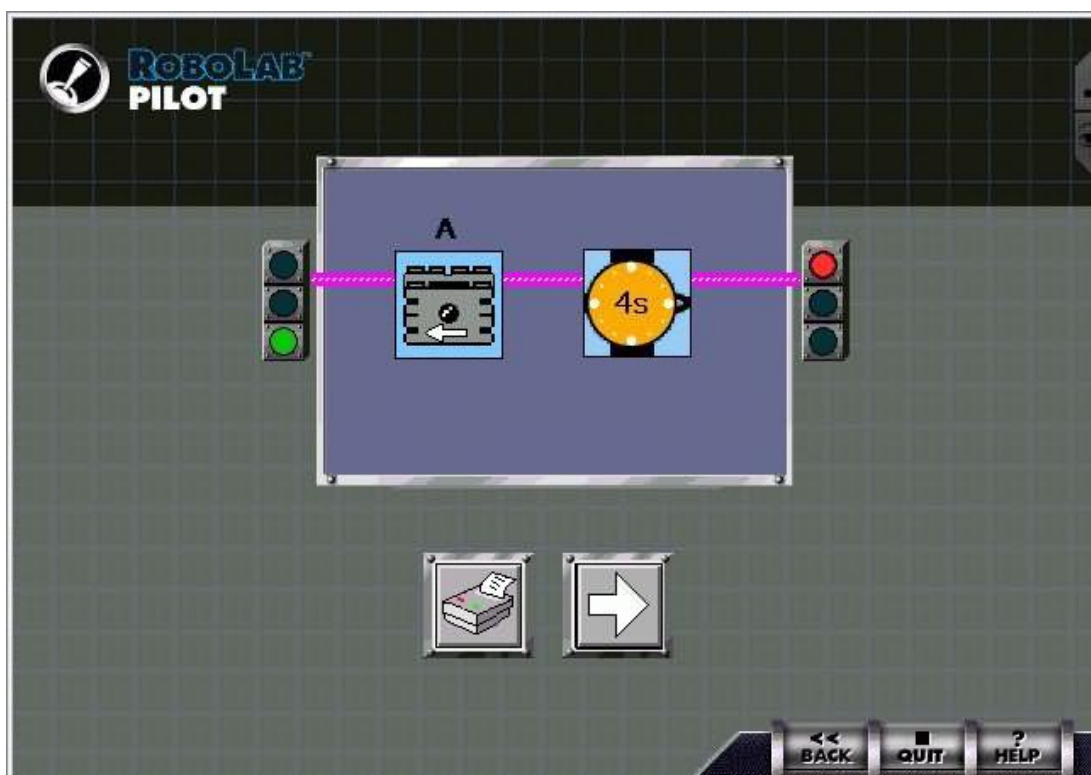
1.4.2 Pracovné prostredie

Pilot

Pilot je programovacie prostredie, ktoré vzniklo na modifikovanie predpripravených programov, šablón. Tento režim bol špeciálne navrhnutý pre celkové oboznámenie so systémom a má umožniť deťom nenásilnú formu výučby

základov algoritmizácie. K tomu je prispôsobené aj prostredie s veľkými ikonami príkazov. Počas práce si väčšinou deti postavlia jedného robota a pomocou prostredia menia hodnoty predpripravených programov a sledujú zmeny. Týmto spôsobom sa oboznámia s programovacím jazykom Robolab.

Pilot je rozdelený do štyroch úrovní podľa obtiažnosti. Najjednoduchší program iba aktivuje motor, zatiaľ čo tie zložité sa rozprestierajú cez niekoľko obrazoviek a obsahujú podmienky a programovanie senzorov. V celkom ôsmich príkladoch tak deti postupne získajú vedomosti a návyky potrebné pre prvé kroky v úrovni Invertor.



Obrázok 6: Jednoduchý Pilot program

Invertor

Invertor je plnohodnotné programovacie prostredie. Ikony jednotlivých príkazov sú menšie, ale dizajn je podobný prostrediu PILOT, takže prechod na Invertor by nemal robiť problém. Programovanie v Invertore prebieha tak, že na

pracovnú plochu programu (okno programu) sa umiestňujú ikony príkazov. Tieto príkazy sa vyberajú z palety príkazov. Paleta sa objaví po kliknutí pravého tlačidla kdekoľvek na ploche. Samotná pracovná plocha programu nemá žiadne obmedzenia, takže sa plynule zväčšuje s tým, ako sa program rozrastá. Veľké programy tým bohužiaľ strácajú na prehľadnosti.

Vedľa palety s príkazmi sa nachádza k dispozícii aj sada nástrojov slúžiacich ku grafickej úprave programu. Zarovnávanie ikon, rozostupov, vkladanie komentárov kamkoľvek v programe je samozrejmosť.

Takisto ako Pilot aj Invertor je rozdelený na štyri úrovne obtiažnosti. S rastúcou úrovňou rastie zložitosť vzorových programov. Spolu so stúpajúcou zložitosťou rastú aj skúsenosti žiakov s programovaním, zväčšuje sa aj počet príkazov ktoré majú k dispozícii na ovládanie RCX jednotky. Pokiaľ v prvej úrovni majú k dispozícii len základnú sadu príkazov pre ovládanie motorov a tlakového senzora, vo štvrtej úrovni sú sprístupnené všetky funkcie vrátane dataloggerov. Pri použití štvrtej úrovne má teda užívateľ úplnú kontrolu nad RCX. Určitou nevýhodou je podľa mňa zhoršená orientácia v takto rozsiahlej ponuke príkazov.

Prostredie Investigator

Predchádzajúce prostredia slúžia hlavne na programovanie a tvorenie samotného programu. Investigator slúži na tvorenie celých projektov. Kľúčovým prvkom Investigatoru je menu. Tlačítka v menu spúšťajú jednotlivé moduly. Samotné programovanie prebieha v prostredí Invertoru, ale programy sa ukladajú ako súčasť projektu. Jazyk je obohatený o príkazy pre prácu s nameranými dátami. Namerané dáta sa dajú pomocou tohto programu odoslať do počítača kde ich môžeme spracovávať a rôznymi spôsobmi vyhodnocovať.

Spracovávanie dát

Na spracovanie nameraných dát je určená oblasť nazývaná Compute area. Dáta môžeme spracovávať na piatich rôznych úrovniach od najjednoduchšieho výpisu až po komplexnú vedeckú prezentáciu. Namerané dáta sa dajú prehliadať, sčítavať rôzne filtrovať a určiť ktoré sa majú zobraziť. To akým spôsobom dáta spracujeme sa znovu zapisuje pomocou vývojového diagramu, podobne ako pri programovaní v prostredí Invertor. Možností ako spracovávať namerané dáta v

Robolabe je obrovské množstvo a ponúkajú spracovanie dát na profesionálnej úrovni. Vedľa integrálov a derivácií nameraných kriviek tu nájdeme aj generátory impulzov sínusových, obdĺžnikových priebehov, rôzne druhy šumu, prekladanie funkcií, alebo štandardné štatistické krivky logaritmické, lineárne, exponenciálne, regresné, analytické.

Program umožňuje dokumentovať projekt a publikovať získané dáta a výsledky v rôznych podobách. Do jednotlivých stránok popisujúcich projekt môžeme vedľa textu a grafov vyprodukovaných priamo v Robolabe vkladať taktiež obrázky, alebo video z pripojenej kamery. Celá prezentácia sa dá potom publikovať formou web prezentácie, prezentácie, alebo jednoducho vytlačiť.

1.5 ISES

Popis programu

Program ISES je štandardný produkt bežiaci v 32 bitových Microsoft Windows. Vďaka spätnej kompatibilite a hardvérovej podpore procesorov dokáže program bežať aj na 64 bitových Microsoft Windows. Jeho ovládanie odpovedá klasickému zaužívanému ovládaniu väčšiny programov (zmena veľkosti okna, presun okna, použitie menu a podobne).

Spustenie programu

Spustenie programu ISES prevedieme ako spustenie akéhokoľvek iného programu v prostredí Microsoft Windows. V prieskumníku súborov nájdeme adresár ISES40 (pokiaľ sme pri inštalácii nezvolili iné meno adresáru) a v nom ďalší podadresár Bin, v nom nájdeme príslušný spúšťačí súbor ISES.exe. Dvojklikom na tento súbor prevedieme spustenie programu.

Ukončenie programu

Program môžeme ukončiť viacerými možnosťami. Prostredníctvom menu programu môžeme zvoliť položku z menu Experiment – Koniec, ktorá beh spustenej aplikácie ukončí. Taktiež môžeme použiť kombináciu klávesov Alt + X, ktorá je skráténím povelom pre výber tejto položky z menu.

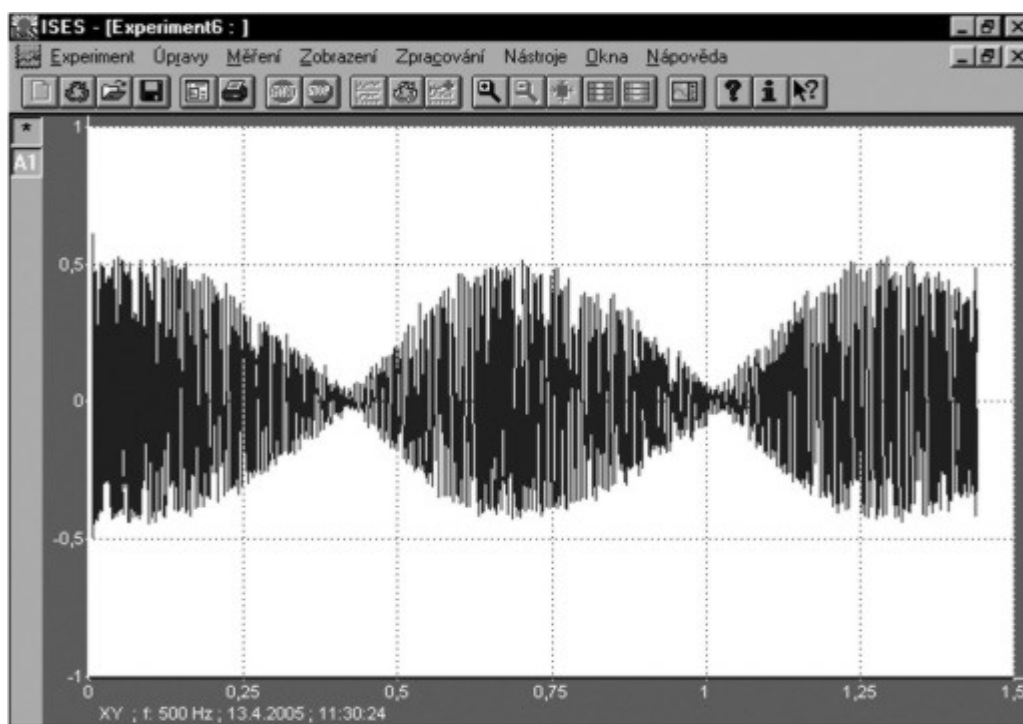
Ďalšou z možností je použitie systémového menu programu, ktoré je spoločné vo všetkých aplikáciách Microsoft Windows, a ukončiť beh programu prostredníctvom neho.

V prípade, ak pri ukončovaní programu ISES pracuje s experimentami, ktoré boli upravované, budeme najskôr vyzvaní k uloženiu zmien v experimente.

1.5.1 Prostredie programu

Hlavné okno programu obsahuje základné ovládacie prvky Windows API: systémové menu, ikony umožňujúce zmenu veľkosti okna, orámovanie, popis, menu. Podľa nastavenia programu je zobrazená aj „lišta ikon“, „lišta s listy experimentu“ a „informačná lišta“.

Hlavné okno programu



Obrázok 7: Okno programu ISES

Systémové menu

Titulok aplikácie, ikony umožňujúce zmenu veľkosti okna, zatvorenie aplikácie.

Menu

Menu programu obsahuje príkazy, ktorými môžeme riadiť činnosť programu ISES.

Lišta ikon

Lišta ikon je umiestnená v hornej časti okna programu a obsahuje najčastejšie používané funkcie programu. Pri kliknutí myšou na ikonu sa vykoná príslušná akcia.

Lišta kariet experimentov

Na tejto lište sú jednotlivé karty s experimentami. Súčasne môže byť spustených viac experimentov a pomocou kariet sa môžeme medzi nimi jednoducho preklikať.

Stavová lišta

Stavová lišta je umiestnená vo spodnej časti okna programu a má informačný charakter.

V jednotlivých poliach lišty sa zobrazuje:

- stručná popis práve realizovanej činnosti
- semafor signalizujúci aktivitu merania zelená, fialová, modrá, červená

Okno programu

V rámci hlavného programu ISES je možné otvárať a spracovávať niekoľko experimentov súčasne. Každému experimentu prislúcha vždy jedno okno.

Základná terminológia

Na usporiadanie programu ISES sa môžeme pozerat' z dvoch uhlov pohľadu:

- ako na meranie a získavanie dat
- z pohľadu zobrazenia prezentovania získaných dat

1.5.2 Meranie a organizácia dat

Experiment

Základnou organizačnou jednotkou programu ISES je experiment. Experiment obsahuje jedno, alebo viac samostatných meraní a je uložený do samostatných súborov na disk počítača. Experiment môže byť ešte „vrstvený“ v jednotlivých kartách experimentu. Každý experiment je potom v programe ISES zobrazovaný v samostatnom okne programu. Výstupný dátový súbor programu potom obsahuje taktiež všetky ďalšie informácie a dáta pre spracovanie, zobrazenie a spracovanie nameraných dát experimentu.

Dáta spracovávané programom ISES, môžeme chápať ako hodnoty veličín, získaných meraním pomocou hardvérových modulov súpravy ISES.

Meranie

Meraním rozumieme dáta, ktoré získame súčasným snímaním a záznamom hodnôt jedného alebo viacerých kanálov merania. Meranie je zvyčajne charakterizované základnými údajmi o frekvencii odpočítaní týchto hodnôt , o dobe trvania merania, o meracích moduloch a podobne. Súčasťou merania môže byť ľubovoľná množina kanálov.

Kanál

Každý modul s ktorým pri meraní pracujeme je pripojený k riadiacemu panelu ISES na určitý kanál. Tieto vstupné kanáli označujeme písmenami A, B, C... tak isto ako na ovládacom paneli súpravy ISES. Ak teda budeme načítavať hodnoty z modulu, ktorý je pripojený v riadiacom paneli súpravy ISES v pozícii B, tak budeme hovoriť, že načítavame hodnoty z kanálu B.

1.5.3 Zobrazenie dát

Okno experimentu

Základná plocha pre zobrazovanie získaných dát pri meraní je okno programu. V rámci tohto okna je možné zobrazovať získané dáta v podobe aká najviac vyhovuje užívateľovi. V okne programu môžeme mať spustených niekoľko kariet s hodnotami, dátami experimentu. V kartách experimentu nemusia byť len hodnoty získané pomocou merania, ale môžeme ich rozšíriť o písaný text, obrázky a ďalší multimediálny obsah.

Panel

Okno programu môže byť rozdelené do niekoľkých samostatných plôch ktoré môžeme pomenovať panele. Ak máme otvorený len jeden panel tak je okno programu vyplnené len ním. V prípade viac otvorených panelov sa okno programu rozdelí a pomocou myši môžeme meniť veľkosť panelu. Každý panel je samostatný a môžeme v nom zobrazovať nezávislé dáta programu.

2. Teoretický popis voľného pádu vo vákuu

Voľne padajúci objekt je objekt, ktorý padá pod vplyvom gravitácie a nepôsobí naň žiadna iná sila. Jedná sa o rovnomerne zrýchlený pohyb. Má smer zvislý nadol so stálym zrýchlením. Veľkosť tiažového zrýchlenia závisí od nadmorskej výšky a od polohy na Zemi. Každý predmet na ktorý pôsobí len gravitačná sila sa nachádza v stave ktorý nazývame voľný pád. Existujú dve dôležité vlastnosti ktoré sú splnené pri voľnom páde:

- na voľne padajúce predmety nepôsobí odpor vzduchu
- všetky voľne padajúce predmety, sú urýchlené rýchlosťou $9,8 \text{ m/s}^2$

2.1 Popis voľného pádu

V okolí Zeme pôsobí na všetky telesá tiažová sila. V blízkosti povrchu Zeme je konštantná. V dôsledku tejto sily padajú telesá smerom k povrchu zeme. Pád telesa pod vplyvom tiažovej sily sa najčastejšie uskutočňuje vo vzduchu.

Voľný pád je pohyb, ktorý teleso koná vplyvom gravitácie vo vákuu. Voľný pád z výšky, ktorá je malá v porovnaní s rozmermi Zeme, je pohyb podobný rovnomerne zrýchlenému pohybu. Zrýchlenie voľného pádu sa nazýva tiažové zrýchlenie a označuje sa g . Na tom istom mieste je pre všetky telesá rovnaké. Mení sa s nadmorskou výškou a zemepisnou šírkou. Preto bola dohodnutá hodnota $g = 9,806 \text{ m/s}^2$ platiaca na zemi, ktorá sa nazýva normálové tiažové zrýchlenie.

Pretože $a = g$, závislosť veľkosti okamžitej rýchlosti voľne padajúceho telesa

od času je vyjadrená vzťahom $v = \frac{1}{2} g \cdot t^2$

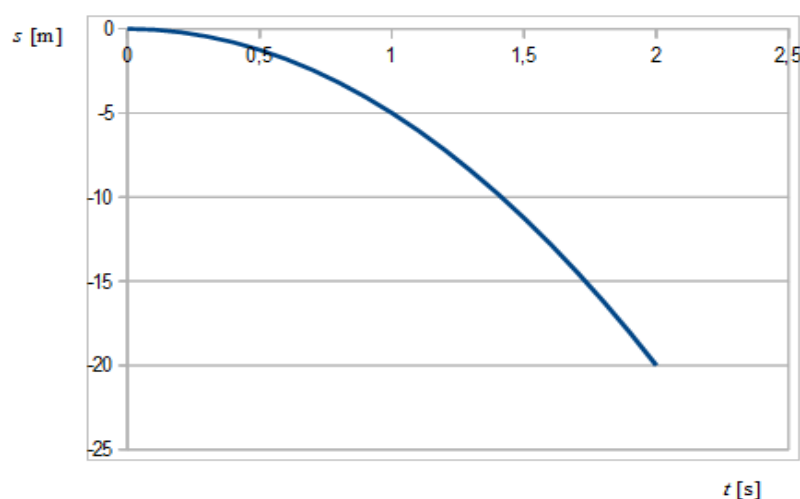
Okrem tiažovej sily vo vzduchu na teleso pôsobí ešte aj odporová sila, v dôsledku odporu prostredia. Veľkosť odporovej sily závisí od rozmerov a tvaru telesa. Aby na teleso nepôsobila odporová sila, muselo by teleso padať vo vákuu. Voľný pád sa nazýva pád volne spusteného telesa vo vákuu s počiatočným zrýchlením $a = 0$.

2.2 Grafické znázornenie voľného pádu

Pri voľnom páde vyšetrujeme tri závislosti a to je závislosť polohy od času $s = s(t)$, závislosť rýchlosti od času $v = v(t)$, závislosť zrýchlenia od času $a = a(t)$. V nasledujúcich grafoch sú zobrazené postupne všetky tri závislosti.

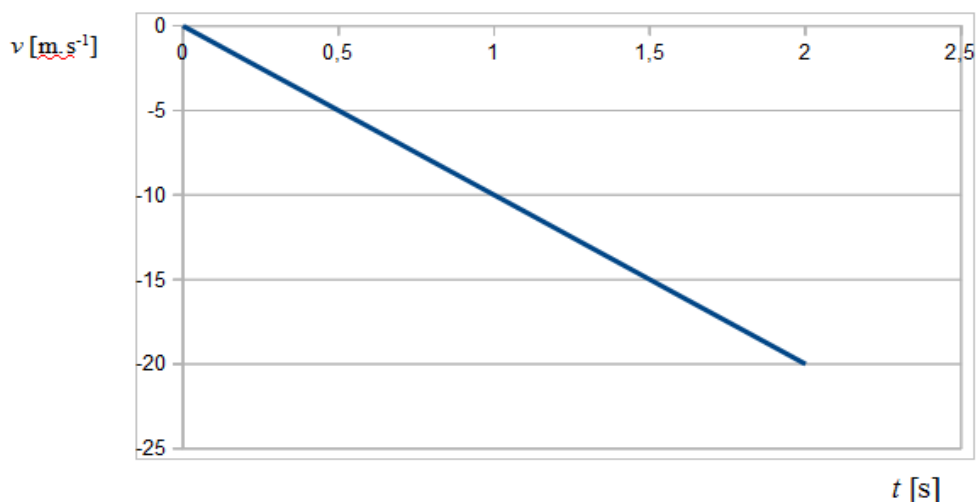
Dráha voľného pádu rastie s druhou mocninou času. Pre dráhu voľného pádu

platí vzťah $s = \frac{1}{2} g t^2$, kde s je dráha, g je gravitačné zrýchlenie, t je čas.



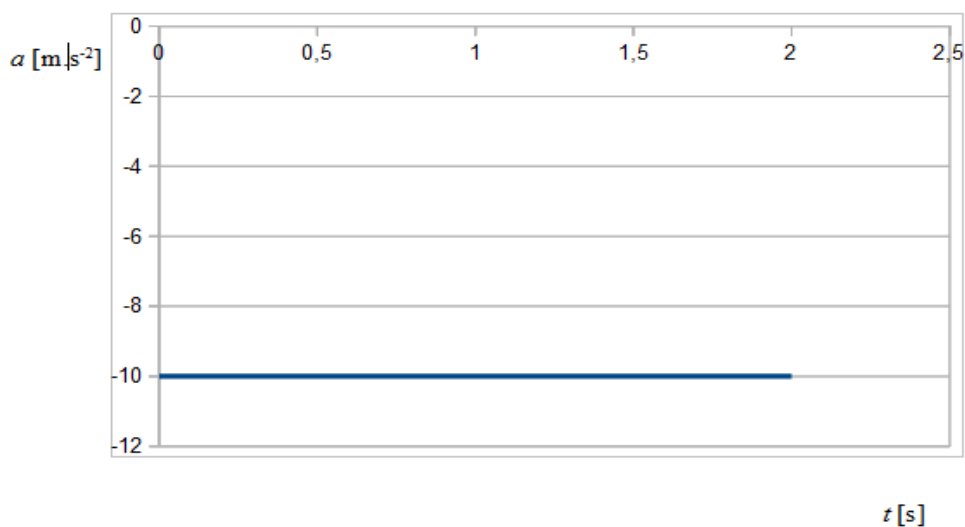
Obrázok 8: Závislosť dráhy od času

Pokusom s Newtonovou trubicou sa dá dokázať, že vo vákuu všetky telesá spustené naraz padajú súčasne. Napríklad, ak z jednej trubice nevyčerpáme vzduch, guľôčka dopadne na dno skôr ako pierko. Pierko má oveľa väčšiu plochu a tým pádom kladie aj väčší odpor vzduchu. Ak z trubice vyčerpáme vzduch, čím dosiahneme vákuum, dopadnú obidve telesá s rovnakým zrýchlením a súčasne.



Obrázok 9: Závislosť rýchlosti od času

Na grafe vidíme závislosť rýchlosti v od času t pri voľnom páde. Rýchlosť narastá priamoúmerne s časom.



Obrázok 10: Závislosť zrýchlenia od času

Ak teleso, ktoré bolo na začiatku v pokoji, uvoľníme a necháme padať voľným pádom, tak jeho zrýchlenie pohybu je $a = -g$. A jeho rýchlosť bude narastať v závislosti na čase $v = g t$. Veľkosť tiažového zrýchlenia závisí od nadmorskej výšky a od polohy na Zemi.

2.3 Matematický popis voľného pádu

Pri voľnom páde dochádza k zrýchleniu padajúceho objektu, pričom jeho parametre pohybu zostávajú nezávislé na hmotnosti padajúceho objektu ak budeme ignorovať odpor prostredia. Normálne, je odpor vzduchu veľmi nízky, s výnimkou objektov s veľkými plochami. Ale zároveň si musíme uvedomiť, že sila ktorú získa objekt pri voľnom páde je určená priamou úmerou hmotnosti objektu. V skutočnosti táto sila objektu nie je nič iné ako hmotnosť objektu a jeho hybnosť.

2.3.1 Rovnice voľného pádu vo vákuu

Podľa druhého Newtonovho zákona je sila potrebná k zmene stavu objektu, a preto sa tento zákon vzťahuje aj na voľný pád. V prípade voľného pádu, je lepšie na označenie vzdialenosti použiť označenie výšky z ktorej je jasný smer. Gravitačné zrýchlenie je medzinárodne dohodnutá konštanta.

Predpokladajme, že 'v' je konečná rýchlosť, 'h' je výška a 't' je čas vzťahujúci sa k voľnému pádu. Je zrejmé, že počiatočná rýchlosť pre voľný pád je 0 a zrýchlenie v tomto prípade je 'g'. Konečná rýchlosť je všeobecne definovaná ako súčet počiatočnej rýchlosti, zrýchlenia a času. Preto pre voľný pád platí:

$$v = 0 + g t$$

Druhá rovnica pre voľný pád sa vypočíta ako vzdialenosť ktorú prejde objekt v každom bode je výsledkom priemernej rýchlosti a času. Preto pri voľnom páde platí:

$$h = \frac{1}{2}(v+0)(t) = \frac{1}{2}vt = \frac{1}{2}gt$$

$$h = \frac{1}{2}gt^2$$

Po odstránení 't' z rovníc dostaneme tretiu rovnicu:

$$h = \frac{1}{2}gt^2 \text{ alebo } \frac{1}{2}gt^2 = h$$

$$\frac{1}{2}g^2t^2 = gh$$

$$\frac{1}{2}v^2 = gh \quad - \text{ (pomocou prvej rovnice)}$$

$$v^2 = 2gh$$

Tieto rovnice sa nazývajú rovnice voľného pádu.

3 Teoretický popis voľného pádu s odporom prostredia

3.1 Matematický popis

V predchádzajúcej kapitole sme odvodili rovnice s uvažovaním pôsobenia gravitačnej sily na objekt. Ale v skutočnosti gravitačná sila nieje jedinou silou ktorá pôsobí na objekt pri voľnom páde. Prejavuje sa tu tiež odpor prostredia. Najčastejšie to je odpor vzduchu. Odpor vzduchu je komplikovaná funkcia a zohľadniť ho rovniciach je dosť komplikované nakoľko to závisí na viacerých faktoroch, ako je teplota, fyzické rozmery objektov, hustota vzduchu a tak ďalej. Našťastie vo väčšine prípadov je odpor vzduchu tak malý, že ho možno zanedbať. V takomto prípade môžeme rovnice odvodené v predchádzajúcej kapitole považovať za pomerne dosť presné.

3.2 Rovnice voľného pádu so zohľadnením odporu prostredia

Zatiaľ som odviedol len rovnice voľného pádu vo vákuu. Teraz sa pokúsím popísať voľný pád zohľadnením odporu prostredia. Sila pri páde musí byť znížená o:

$$F = m g - k v^2$$

Kde 'm' je hmotnosť objektu a 'k' je konštanta, označovaná ako koeficient odporu pre konkrétny typ pohybu. V tomto prípade vzorce voľného pádu nie sú tak jednoduché a musíme použiť diferenciálne operácie.

$$v(t) = A \tanh\left(\frac{t}{B}\right)$$

$$A = \sqrt{\frac{mg}{k}} \quad \text{a} \quad B = \sqrt{\frac{m}{gk}}$$

3.3 Odvodenie vzorcov

Odporová sila úmerná druhej mocnine rýchlosti. Táto závislosť je charakteristická pre pohyb telesa v plyne. Odporová sila musí mať smer opačný, ako je smer rýchlosti z čoho vyplýva: $m \dot{\vec{v}} = \vec{F} - K \vec{v} \cdot \vec{v}$ symbol v bez šípky znamená veľkosť rýchlosti. Najskôr odvodím vzťahy pre priamočiary pohyb, $\vec{v} \parallel \vec{F}$.

Potom môžeme rovnice prepísať do skalárneho tvaru $\dot{v} = A^2 - B^2 v^2$, kde

$$A^2 = \frac{F}{m}, \quad B^2 = \frac{K}{m}. \quad \text{Ak budeme predpokladať } A \neq 0, \text{ tak rozkladom na}$$

čiastočné zlomky získame $\frac{\dot{v}}{A+Bv} + \frac{\dot{v}}{A-Bv} = 2A$, z čoho po integrovaní

dostaneme $\ln \frac{A+Bv}{A-Bv} = 2(ABt+C)$. Z tohto vzorca môžeme priamo vyjadriť

rýchlosť ako funkciu času $v = \frac{A}{B} \tanh(ABt+C)$.

Vidíme tu, že rýchlosť rastie od počiatočnej hodnoty v_0 a blíži sa k inej rovnovážnej hodnote, v tomto prípade $v_1 = A/B$. Integračná konštanta C sa určuje s počiatočných

podmienok. Ak je pri $t = 0$ rýchlosť $v = v_0$ musí platiť $v_0 = \frac{A}{B} \tanh(C)$, alebo

$C = \frac{1}{2} \ln \frac{A+Bv_0}{A-Bv_0} = \frac{1}{2} \ln \frac{v_1+v_0}{v_1-v_0}$. Po integrovaní vzťahu nájdeme dráhu telesa

$s - s_0 = \frac{1}{B^2} \ln \frac{\cosh(ABt+C)}{\cosh(C)}$. Ak platí $v_0 = 0$ pri voľnom páde, tak $C = 0$ a

pre rýchlosť dostaneme vzťah $v = v_1 \tanh(ABt)$ a pre dráhu

$$s = \frac{1}{2} \ln \cosh(ABt) .$$
 Ak by platilo $A - Bv_0 = 0$, teda $v_0 = v_1$ tak zrýchlenie by

bolo od počiatku rovné nule a jednalo by sa o rovnomerný priamočiary pohyb.

4 Voľný pád v systéme ISES

Pomocou systému ISES som zmeral voľný pád v sklenenej trubici vo vákuu a voľný pád v sklenenej trubici. Výsledky som zaznamenal a merania vyniesol do grafov.

4.1 Popis experimentu

Meraný experiment pozostáva z dvoch sklenených trubíc ktoré sú dlhé 1 m. Tieto trubice majú po celej dĺžke rovnaký priemer 0,01 m. Na povrchu každej trubice sú navinuté závit, ktoré vytvárajú 10 medených cievok umiestnených v rovnakých rozostupoch od seba. Tieto cievky sú zapojené do série a následne cez modul voltmetra sú prepojené so systémom ISES. Pri prechode kovového telesa vnútom cievky sa indukuje napätie ktoré sa vyhodnotí pomocou pripojeného systému ISES. Pri experimente sledujeme pád magnetu ktorý sa spúšťa v trubici ktorá je na konci otvorená a pád magnetu v trubici z ktorej je pomocou vývevy odsatý vzduch. Celi zariadenie je napojené na počítač v ktorom sa pomocou systému ISES spracovávajú a vyhodnocujú získané dáta. Grafickým výstupom je závislosť okamžitého indukovaného napätia ako funkcie času.

4.2 Princíp experimentu

Experiment voľného pádu telesa v sklenenej trubici vo vákuu a s odporom prostredia pracuje na princípe Faradayovho zákona o indukovanom elektromotorickom napätí určeného vzťahom

$$U = \oint E ds = - \frac{d\Phi}{dt} .$$

Tento vzťah definuje, že elektromotorické napätie indukované v uzavretej krivke sa rovná zápornej časovej derivácii indukčného toku plochou ohraničenou danou uzavretou krivkou [Krempaský a Schauer, 2006]. Pri voľnom spustení magnetu do

trubice magnet padá nadol a prechádza cez jednotlivé cievky umiestnené na trubici. Pri prechode cez cievku, v dôsledku zmeny magnetického poľa sa v nej naindukujú napätie, ktoré je v systéme ISES zaznamenané na osi y. Os x je časovou osou, graf je závislosť napätia od času. $U = U(t)$. Ak budeme vychádzať z predchádzajúcej rovnice pre magnetický tok tak dostaneme rovnicu

$$\Phi = -\frac{\mu_0 m N}{2} \frac{b^2}{(b^2 + z^2)^{\frac{3}{2}}}.$$

Taktiež môžeme odvodiť vzťah pre veľkosť okamžitého elektromotorického napätia

$$U = -\frac{d\Phi}{dt} = -\frac{d\Phi}{dz} \frac{dz}{dt} = -\frac{d\Phi}{dz} v = \frac{3\mu_0 m N b^2}{2} \frac{zv}{(b^2 + z^2)^{\frac{5}{2}}}.$$

Kde N je počet závitov cievky, m magnetický moment, b polomer cievky, z vzdialenosť magnetu od stredu cievky táto hodnota môže nadobudnúť kladné aj záporné hodnoty a v rýchlosť padajúceho magnetu. Z predchádzajúcej hodiny vyplýva, že veľkosť okamžitého indukovaného napätia U pri rozmeroch použitej trubice závisí od rýchlosti pohybujúceho sa magnetu vnútri trubice a od vzdialenosti magnetu od stredu cievky. Ak z trubice odčerpáme všetok vzduch bude veľkosť rýchlosti magnetu narastať podľa vzťahu $v = g t$.

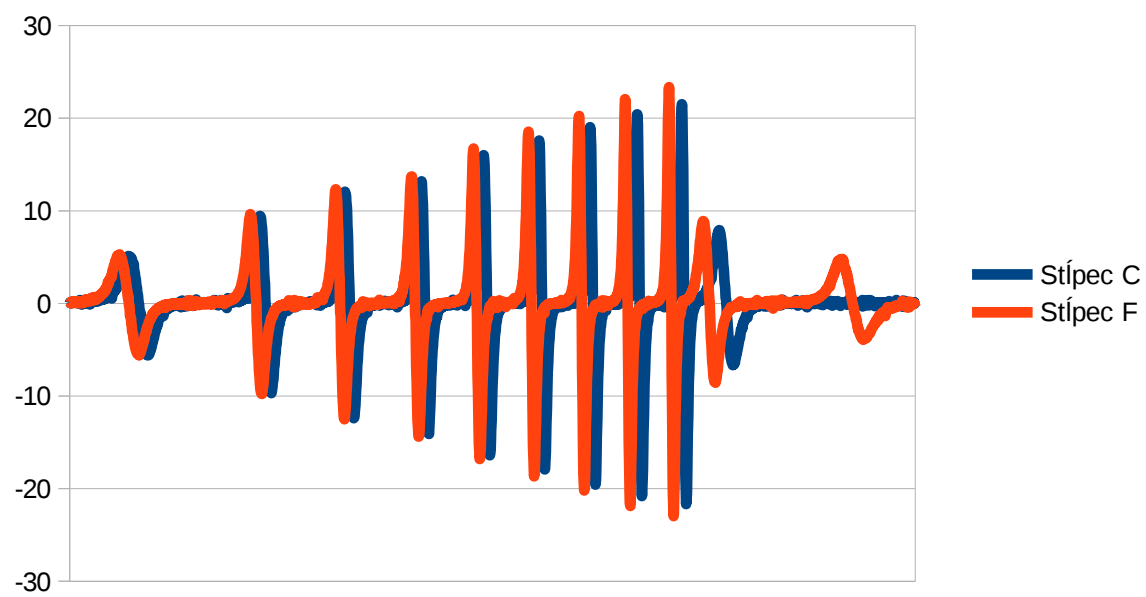
V reálnej situácii však na padajúci objekt pôsobia odporové sily a rýchlosť pohybujúceho sa magnetu môžeme vyjadriť riešením diferenciálnej rovnice

$$m \frac{d^2 y}{dt^2} = mg - kv.$$

So všeobecným riešením pre pohyb v plynnom prostredí s nízkym tlakom vztlakovú silu zanedbávame

$$v(t) = \frac{mg}{k} \left(1 - e^{-\frac{kt}{m}} \right).$$

Kde m je hmotnosť padajúceho telesa, t čas, v rýchlosť telesa a k je koeficient dynamického trenia.



5 Simulácia voľného pádu v prostredí EJS

Modelovanie reálnych dejov pomocou počítača sa stáva na školách populárnym doplnkom pri výklade nových tém. Modelovanie sa uplatňuje hlavne v prírodovedných vedách. Uchýľujeme sa k nemu väčšinou vtedy, keď je prevedenie reálneho experimentu z nejakého dôvodu nereálne, alebo veľmi ťažké. Ďalšou významnou oblasťou využitia je predpovedanie a prípadné optimálne ovplyvnenie prírodných dejov. Napokon z hľadiska výučby, podobne ako väčšina pomôcok a nástrojov pri rozumnom uvažovaní a primeranom zapojení do výučby, alebo cvičení umožňuje hlbšie pochopenie zákonitostí. Pri tom sa opiera o prirodzený záujem žiakov a študentov o výpočtovú techniku.

5.1 Popis prostredia EJS

Klasické programovacie jazyky ako sú napríklad C++, Pascal, C, C#, alebo Java vyžadujú zvládnutie ich syntaxe a pomocných nástrojov na profesionálnej úrovni. Preto sú cez všetky svoje výhody, pre bežné nasadenie vo výučbe nevyhovujúce a náročné na zvládnutie. Našťastie sa objavujú softvérové prostredia, u ktorých sa aspoň pri základnom použití zaobídeme bez pokročilej znalosti programovania. Takéto softvérové riešenia umožňujú užívateľovi sústrediť sa na samotný matematický model modelovaného, študovaného deja.

Balíček Easy Java Simulations vyvinul matematik Francisco Esquembre z University of Murcia v Španielsku. Tento balík dal voľne k dispozícii všetkým záujemcom pod licenciou GPL. Je voľne dostupný na jeho webovej stránke. Nadväzuje na projekt Open Source Physics, v rámci ktorého bola vytvorená kompletná sada knižníc, tried a nástrojov pre programátorov na vytváranie vlastných fyzikálnych modelov v jazyku Java. Tento projekt je taktiež uvoľnený pod licenciou GPL3 a je dostupný zadarmo. Bez skúseností s programovaním, je využitie prostriedkov Open Source Physics ťažké. Easy Java Simulation je tak medzistupňom, ktorý sprístupňuje knižnice Open Source Physics aj začiatočníkom bez znalosti programovacieho jazyka. Podľa dokumentácie programu predstavuje balík EJS v

podstate vývojový softvér, ktorý dokáže využívať knižnice Open Sourc Physics. Umožňuje rýchlo a jednoducho prepísať matematický model do podoby spracovateľnej počítačom. Následne tento model otestovať a vytvoriť spustiteľnú Java simuláciu prípadne webstránku s Java Appletom.

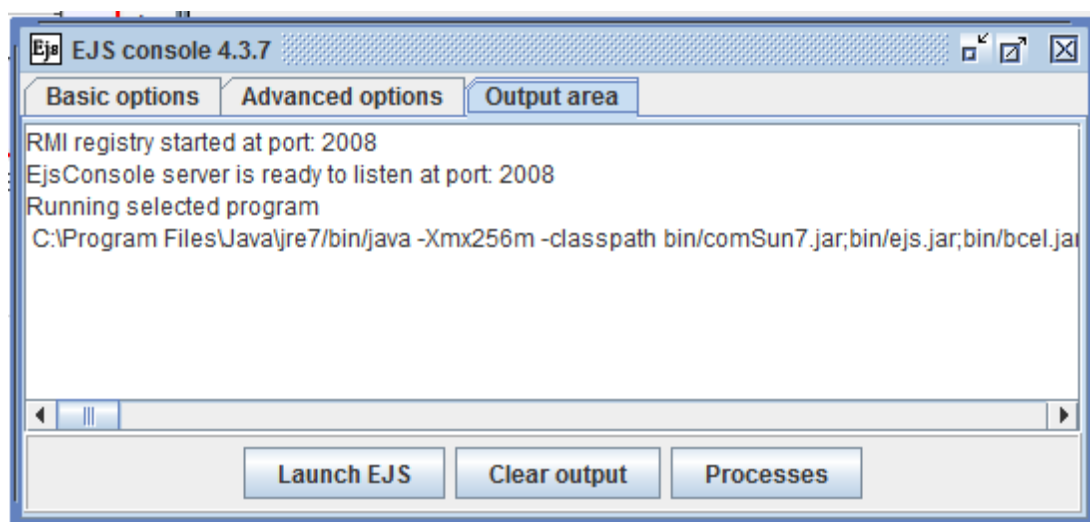
Taktiež inštalácia EJS nevyžaduje špeciálne zručnosti užívateľa. Inštalačný ZIP archív si môžeme zadarmo stiahnuť z webovej stránky, skopírovať do zvoleného adresára a spustiť súborom EjsConsole.jar. Program pre svoj beh vyžaduje Java Runtime Enviroment verzie 1.6.0 a vyššie. Prostredie JRE si užívatelia môžu stiahnuť zo stránok Oracle, je poskytované bezplatne. Záujemcovia o pokročilejšie programovanie v jazyku Java budú potrebovať Java Development Kit.

K ukladaniu vytvorených modulov a ich zdieľanie medzi užívateľmi využíva program Easy Java simulatin syntax XML, s ktorou sa však užívatelia vďaka grafickému rozhraniu nemusia zaoberať. Primárne XML súbory slúžia na uchovávanie všetkých potrebných informácií k vytvoreniu simulácie, je ich však potrebné nakopírovať do pracovného adresára zadefinovanom pri prvom spustení programu Easy Java Simulations. Zdieľať môžeme samozrejme aj spustiteľné exportované JAR súbory, alebo webové stránky a java appletmi. Pri takomto zdieľaní máme obmedzené možnosti ďalšieho ladenia výpočtovej časti modelu. Základná dokumentácia je súčasťou programu. Zo stránok projektu EJS sa taktiež dá stiahnuť veľké množstvo hotových simulácií čím sa môžeme učiť na príkladoch skúsených kolegov. Podrobnejší opis prostredia môžeme nájsť na stránke autora projektu.

Ako väčšina softvéru má aj Easy Java Simulations svoje výhody a nevýhody, ktoré môžu mať pre každého užívateľa inú dôležitosť. Môžem poukázať na obmedzenú oblasť využitia numerického riešenia pohybových rovníc, malá možnosť ovplyvnenia výsledného vzhľadu grafického okna programu. Do značnej miery skrytý vlastný algoritmus numerického výpočtu. Podľa môjho názoru však prevažujú výhody ako sú intuitívne grafické prostredie, prenositeľnosť medzi rôznymi operačnými systémami ako sú Windows, Mac Linux. Dostupnosť programu zaistená licenciou GPL čo zaručuje dostupnosť projektu zadarmo, podpora open source komunity, možnosť modifikovania prostredia a v neposlednom rade možnosť exportu na webstránky.

Súčasť prostredia EJS

Po spustení EjsConsole.jar sa otvorí EjsConsole, v ktorom môžeme vykonávať základné nastavenia, a v druhom samostatnom okne sa potom spustí samostatný program. Nájdeme tu panel nástrojov, dolný box na správy a chybové hlásenia a tri hlavné záložky Description, Model, View, s ktorými pracujeme pri vytváraní simulácie.



V panely nástrojov nájdeme ikony pre vytvorenie novej simulácie, taktiež otvorenie rozpracovanej simulácie a uloženie výslednej simulácie. Ďalej tu nájdeme dôležitú ikonu Run, ktorá slúži k spúšťaniu práve otvorenej simulácie a zároveň vytvára spustiteľný Java applet. Pre vytvorenie Java appletu a spustenie programu môžeme tiež zabezpečiť ikonou Edit options. Po kliknutí na ikonu môžeme nastaviť niektoré kľúčové vlastnosti výslednej simulácie vrátane formy výstupu. Ostatné ikony nie sú pre základnú prácu nevyhnutne dôležité. V okne pre správy a chybové hlásenia nám program hlási základné informácie, ako je úspešné uloženie súboru, alebo naopak. Ak program nemôže uložiť súbor tak vypíše chybu a stručný popis prečo nemôže vytvoriť výslednú simuláciu. Uprostred programu Easy Java Simulation dominuje pracovná plocha, do ktorej zadávame vstupné dáta, z ktorých program vytvorí konečnú simuláciu. Do tejto oblasti zadávame aj stručný popis simulácie. Tento popis sa zobrazuje aj v exportovaných html stránkach ako Description. Na inom mieste sa definujú používateľské premenné a zadávajú rovnice, ktorými sa má simulácia riadiť Model. V neposlednom rade vytvárame konečnú

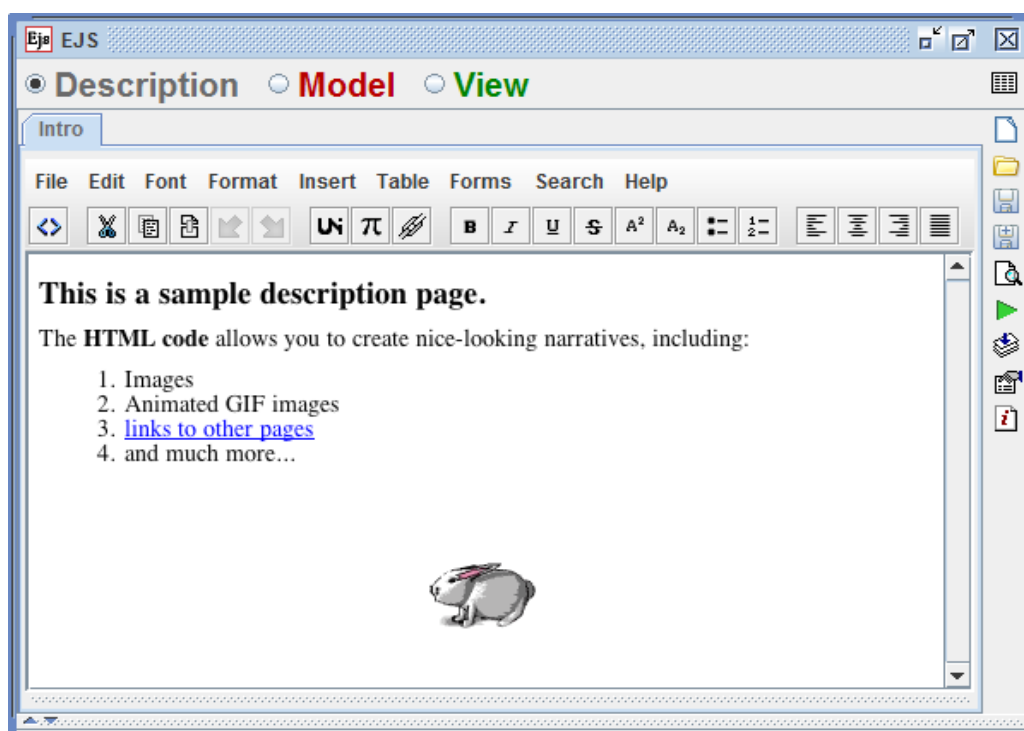
výstupnú grafickú podobu programu View. Podrobnejší popis týchto častí programu popíšem v ďalšej časti práce.

5.2 Popis panelov EJS

5.2.1 Description panel

Panel Description nám poskytuje miesto, kde môžeme popísať tvorenú simuláciu. Popis môže obsahovať vzorce, obrázky, vysvetlenie simulácie. Táto informácia sa bude zobrazovať užívateľom pri štarte, keď bude simulácia spustená ako aplikácia, alebo bude súčasťou html stránky kde bude bežať ako Java applet. Je preto dôležité venovať sa tomuto popisu a správne naformátovať text. Existujú tri spôsoby ako napísať popis.

Prvý z nich je použitie jednoduchého html editoru ktorý je súčasťou programu EJS. Tento editor funguje v dvoch režimoch, čo vidíme to dostaneme ako výstup, alebo sa môžeme prepnúť do režimu zdrojového kódu, kedy sa zobrazuje priamo zdrojový kód html ktorý môžeme následne editovať.



Obrázok 12: Panel Description EJS

Ak však dávate prednosť inému HTML editoru, môžete si text naformátovať v ňom a následne celý HTML súbor naimportovať do programu EJS. V tomto prípade, klikneme pravým tlačítkom myši na prázdny popis panelu, alebo na karte existujúcej stránky a z popup ponuky vyberieme import HTML súboru.

Možnosť importovať HTML stránku. Táto voľba načíta existujúcu HTML stránku a začlení ju ako súčasť simulácie EJS. Účinok bude rovnaký ako keby sme kompletný popis napísali v HTML editore EJS.

Odkaz na externú stránku HTML. Pri tejto voľbe sa popis zobrazí ako súčasť simulácie EJS avšak nebude súčasťou simulácie. EJS aplikácia nahrá popis pri každom spustení simulácie do pamäte. Ak sa však prepojenie na tento popis zmení, alebo sa zmení umiestnenie externého HTML súboru môže sa stať, že popis simulácie nebude dostupný.

5.2.2 Model panel

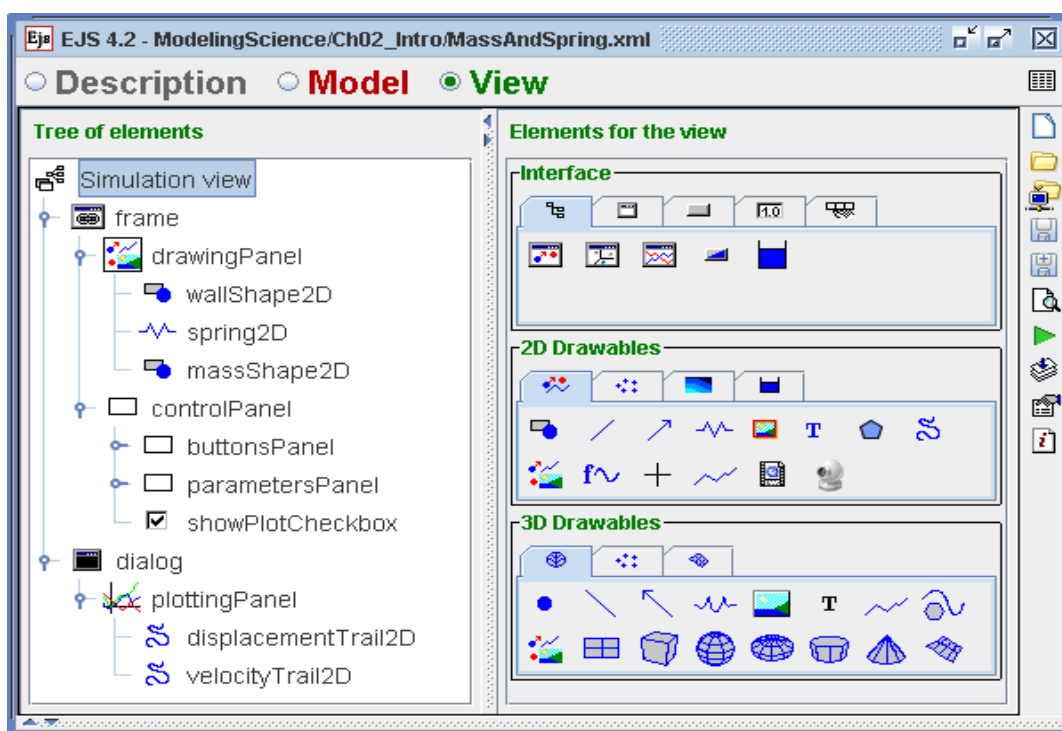
V rámci model panelu môžeme nájsť šesť subpanelov. Premenné, Inicializácia, Evolution, Pevné vzťahy, Customs a Prvky.

- Premenné sa používajú na deklarovanie a inicializovanie premenných modelu.
- Model inicializácia umožňuje zadať ďalšie inicializačné algoritmy.
- Model Evolution opisuje, ako sa model vyvíja v čase. To zahŕňa
 - Editor EDS v programe EJS.
 - Stránku na zapisovanie kódu v editore ODE.
 - EJS a ODS riešenia a malý teoretický popis.
 - Podpora diferenciálnych rovníc.
 - Akcie.
 - Priama kontrola EDE riešenia.
- Model Pevné vzťahy stanovuje pevné vzťahy medzi premennými a ďalšími výpočtami potrebnými na aktualizovanie modelu.
- Model Customs umožňuje autorovi projektu vytvoriť metódy, funkcie a podprogramy pre použitie v iných častiach modelu. Tieto metódy môžeme pridať do knižnice programu EJS ako preddefinované metódy.
- Model Elements ponúka preddefinované a pripravené objekty ktoré môžeme okamžite použiť v simulácii EJS. Tieto objekty slúžia na zjednodušenie niektorých pokročilých úloh.

Kliknutím na subpanel Interior vytvoríme novú stránku, ktorú musíme pomenovať. Na takto vytvorenú stránku ukazuje záložka v hornej časti panela. Kliknutím pravým tlačidlom myši na záložky umiestnené v hornej časti okna sa otvorí rozbaľovacia ponuka. Ponúka nám možnosti k vytváraniu nových záložiek a presúvanie stránok tak, aby sme si mohli premenné zoradiť prehľadnejšie. Je možné povoliť, alebo zakázať niektoré stránky. Toto je dobré pri ladení simulácie, kedy môžeme určité funkcie zakázať.

5.2.3 View panel

Tretí Easy Java Simulations panel je View. Tento panel nám umožňuje vytvoriť grafické rozhranie, ktoré zahŕňa vizualizáciu, interakciu užívateľov a kontrolné prvky programu s minimálnymi znalosťami programovania. Na pravej strane panelu sa nachádzajú jednotlivé elementy ktoré nám program EJS ponúka. V ľavej strane panelu je stromová štruktúra pomocou ktorej vytvárame samotný vzhľad okna simulácie.



Obrázok 13: View panel EJS

V tomto prípade na obrázku 13 vidíme návrh grafického prostredia programu simulácie v EJS. Výstupná grafická aplikácia obsahuje dve okná, rámy a dialóg. Tieto prvky patria do triedy kontajnerových prvkov, ktorých primárnym cieľom je vizuálne zoskupiť ostatné ovládacie prvky simulácie v používateľskom rozhraní programu. Strom zobrazuje popisné názvy prvkov a ikony pre tieto prvky. Prvky sa pridávajú do stromovej štruktúry y ponuky pomocou funkcie drag and drop. Kliknutím na prvok v ponuke sa kurzor zmení na čarovnú paličku. Prvok následne premiestnime do stromovej štruktúry kde ho vhodne umiestnime s ohľadom k nadradenému prvku. Do značnej miery stromová štruktúra určuje kde bude prvok

umiestnený vo výstupnom grafickom zobrazení. To sa dá ovplyvniť taktiež možnosťami rozloženia prvku. Prvky sú rozdelené do troch skupín.

- Prvky ktoré ovplyvňujú modelové premenné a vlastnosti simulácie
- Prvky ktoré ovplyvňujú vlastnosti výrazov
- Prvky referenčné

Vlastnosti prvkov môžeme meniť a rozširovať pomocou vlastných definovaných funkcií a podprogramov. Na výber máme aj veľké množstvo už preddefinovaných funkcií ktoré nám EJS ponúka. Pri návrhu prostredia musíme dávať pozor na umiestnenie prvkov v stromovej štruktúre.

5.3 Popis simulácie

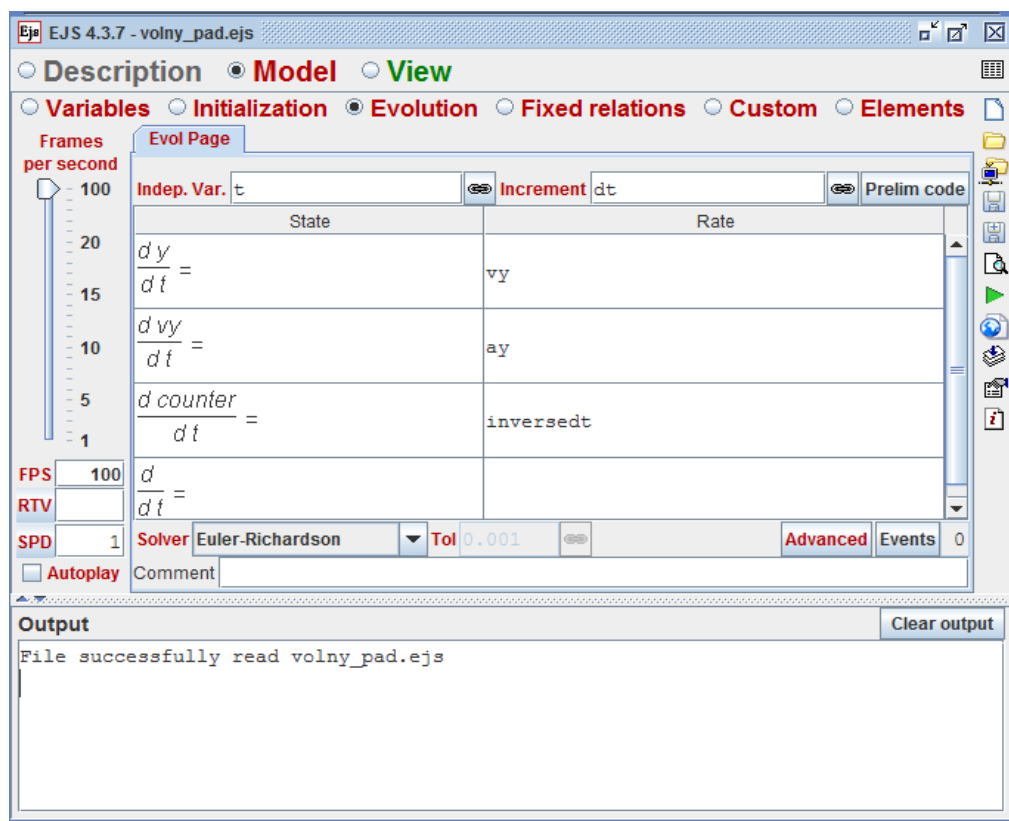
Z fyzikálneho hľadiska sa jedná o známu problematiku voľného pádu. Pri páde na objekt pôsobia odporové sily a polohu pohybujúceho sa objektu môžeme vyjadriť diferenciálnou rovnicou.

$$m \frac{d^2 y}{dt^2} = mg - kv$$

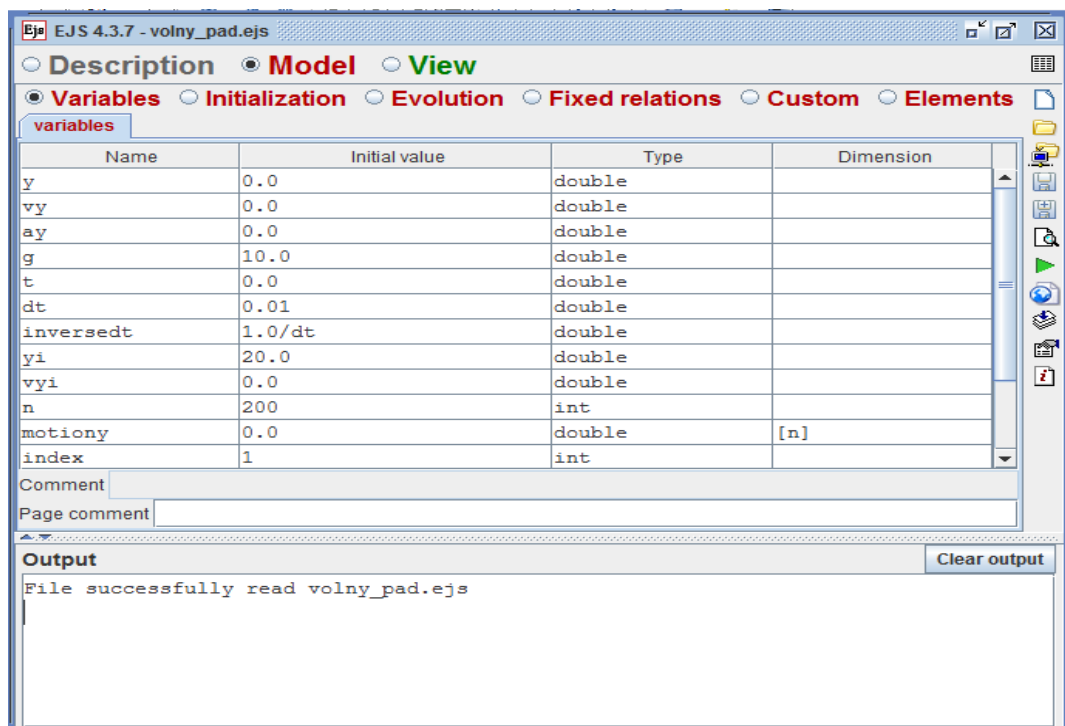
Túto rovnicu môžeme upraviť a dostaneme diferenciálnu rovnicu prvého rádu z ktorej môžeme určiť polohu objektu v čase.

$$\frac{dy}{dt} = v_y \quad \frac{dv_y}{dt} = a_y$$

V prostredí EJS potom tieto rovnice prepíšeme do časti Model/Evolution, kde taktiež zvolíme integračnú metódu. Na výber máme 8 metód ako sú Eulerova, Rungeova – Kuttova štvrtého rádu a podobne. Nastavíme vlastnosti simulácie ako sú tolerancia, automatické spúšťanie appletu a podobne. Taktiež je potrebné zadať všetky premeny v časti Model/Variables. K rovnici pripojíme potrebné moduly pre vykresľovanie animácie padajúceho telesa View a previažeme ich s jednotlivými zadanými premennými. Po týchto krokoch môžeme model simulácie uložiť a pomocou tlačítka Run simulation spustiť. Ak je všetko v poriadku a nenastali žiadne komplikácie, môžeme sledovať animáciu.

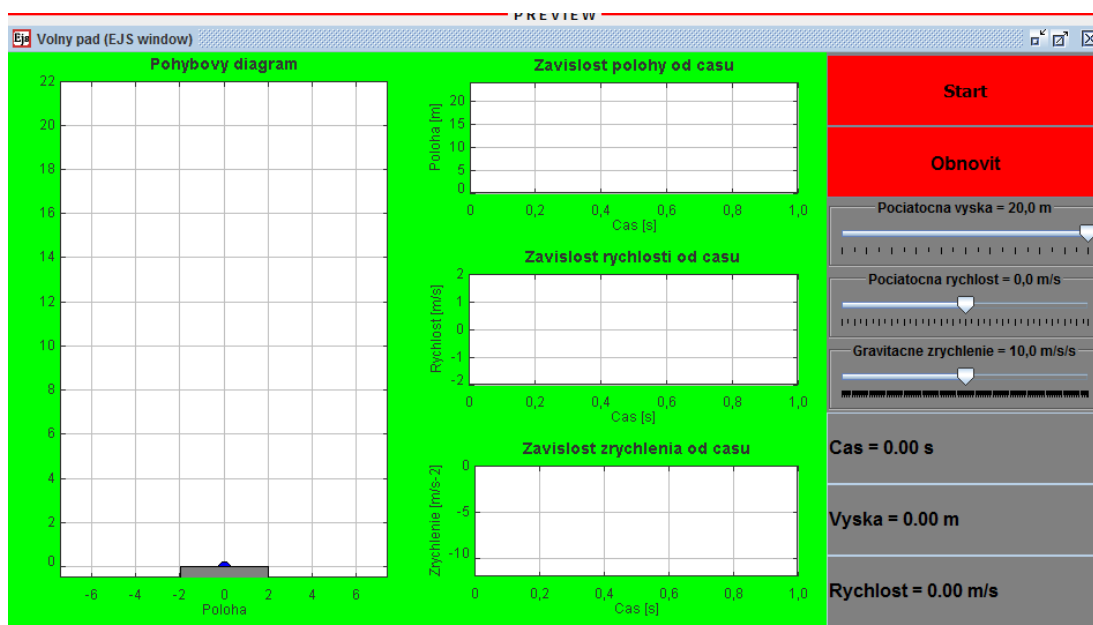


Obrázok 14: Zadávanie vzorcov v EJS



Obrázok 15: Definovanie premenných v EJS

Výstupné hodnoty môžeme meniť, alebo priamo zadávať pomocou zapísania hodnôt. Taktiež tieto hodnoty môžeme zmeniť pomocou grafických možností ako sú posuvníky a podobne.



Výsledná simulácia obsahuje animáciu voľného pádu. Vykresľuje tri závislosti a to závislosť polohy od času, závislosť rýchlosti od času a závislosť zrýchlenia od času. Taktiež umožňuje zmenu hodnôt počiatočnej výšky, počiatočnej rýchlosti a gravitačného zrýchlenia. Obsahuje dva tlačítka slúžiace na spustenie, zastavenie a tlačítko na reštart animácie. Pri behu simulácie sa zobrazujú informácie čas, výška a rýchlosť. Simulácia vytvorená v programe Easy Java Simulation je súčasťou diplomovej práce a je priložená v prílohách. Táto simulácia sa dá využiť vo vyučovacom procese ako názorná pomôcka na pochopenie problematiky voľného pádu.

Záver

Bibliografia

- [1] Bueche, F.J.: Principles of Physics. New York: The McGraw-Hill Co. Inc. 1988.
ISBN 0-07-303579-3
- [2] Fuka, J. - Lepil, O. - Bednařík, M.: Konkrétní didaktika fyziky na střední škole.
Část A. Olomouc: R UP v Olomouci. 1983
- [3] ISES Uživatelská příručka (pracovní verzia 1.11.2006)
- [4] Christian W., Esquembre F.: „Modeling Physics with Easy Java Simulations“.
Phys. Teacher 45 (2007), s. 475.
- [5] Lepil O., Richterek L.: Dynamické modelování. Repronis, Ostrava 2007
- [6] Bajer J.: *Mechanika 2*. UP Olomouc 2004.
- [7] Urquía A., Martín-Villalba C.: Virtual-lab implementation with EJS
- [8] Turek, I. (2009). Didaktika. Bratislava, Iura Edition
- [9] Schauer, F., Lustig, F. and Ožvoldová, M. (2009). ISES - Internet School
Experimental System for Computer-Based Laboratories in Physics. INNOVATIONS
2009, World Innovations in Engineering Education and Research, iNEER. Special
Volume 2009: 109-118.
- [10] Krempaský, J. a Schauer, F. (2006). Fyzika Elektřina a magnetismus. Trnava.
- [11] Driensky, D. a Hrmo, R. (2004). Materiálne didaktické prostriedky. Bratislava,
MtF STU Bratislava.