

# FIZIKAI SZIMULÁCIÓK MEGVALÓSÍTÁSÁNAK EGY ESZKÖZE: AZ EJS KERETRENDSZER

## A TOOL FOR REALIZING PHYSICAL SIMULATIONS: THE EJS FRAMEWORK

Gál Tamás Zoltán<sup>1</sup>, Bérces György<sup>2</sup>

<sup>1</sup>Q SX Kft., Budapest

<sup>2</sup>Eötvös Loránd Tudományegyetem, TTK, Anyagfizikai Tanszék, Budapest

### ÖSSZEFOGLALÁS

*Napjainkban az információ technológiai eszközök a középiskolai tanárok számára is elérhetők. Egy ilyen eszköz az Easy Java Simulation (EJS) keretrendszer. Ez különösen alkalmas fizikai szimulációk készítésére. Mivel az EJS használata viszonylag egyszerű, segítségével az információ-technológiában jártas fizika tanárok is képesek az óráikon használatos alkalmazásokat írni. Az EJS keretrendszer főbb tulajdonságainak ismertetése után bemutatjuk az általunk fejlesztett programokat. Ezek közül egyik a merőleges rezgések összetevésének bemutatására szolgál (Lissajous görbék). A másik program pedig lefényképezett síkbeli testek nevezetes fizikai jellemzőit (tömegközéppont, tehetetlenségi nyomaték, fő tengelyek iránya) képes meghatározni.*

### ABSTRACT

*Nowadays the tools provided by information technology are also available for high school teachers. One of these tools is the Easy Java Simulation (EJS). This program shell is extremely helpful in writing simulation programs in physics. EJS is relatively easy to use, so a physics teacher - who is familiar with information technology - is able to write simple applications for their lectures. After summarizing the main properties of the EJS shell, we present our own, recently developed programs. One of these is a simulation of Lissajous curves. The other program calculates several fundamental mechanical parameters (center of mass, moment of inertia, principal axes) for rigid bodies.*

### KULCSSZAVAK/KEYWORDS

Számítógép, szimuláció, EJS

Computer, simulation, EJS

### SZIMULÁCIÓK AZ ISKOLÁBAN

Szimulációk iskolában való bemutatásához több feltételnek is teljesülnie kell [1]:

- Alapvető követelmény a megfelelő technikai háttér.
- A szimulációk alkalmazása a tanártól informatikai kompetenciát és időráfordítást igényel.

- Egy jó szimuláció nem több mint egy segédeszköz. Nem szabad túl sok információt megjeleníteni a képernyőn, mert a program célja nem a teljesség. Helytelen az is, ha a program használata indokolatlanul bonyolult.
- A jó szimuláció egymás után *többször is lejátszható*. Ezáltal behatóbban megfigyelhetjük a jelenséget, mozgást.
- Fontos feltétel, hogy a szimuláció lehetőleg valósághű legyen! Például ne elégedjünk meg mozgások szimulációjánál az egymást követő ugráló képekkel, hanem tűnjék a mozgás időben folytonosnak.
- **Az animáció, szimuláció hasznos segédeszköz a tanításban. Nem pótolja azonban az elvégezhető kísérletet!**

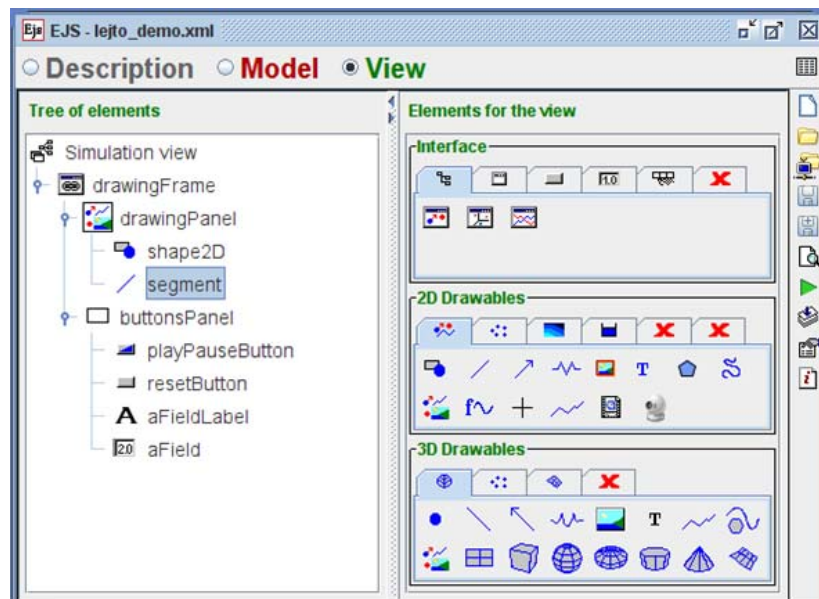
### AZ EJS KERETRENDSZER

Az ezredforduló után jelent meg az EJS nevű program, amelynek deklarált célja az, hogy egy jól használható keretrendszert nyújtson fizikailag korrekt szimulációk elkészítéséhez [2]. Fejlesztését *Francisco Esquembre*, a Murcia-i Egyetem oktatója végzi, az *Open Source Physics* projekt keretében [4]. Az alkalmazás főbb jellemzői:

- **Easy**, azaz egyszerű. A fejlesztők nagy figyelmet fordítottak arra, hogy a program használata könnyen elsajátítható legyen. A leginkább megcélzott felhasználók a fizika tanárok. tőlük pedig nem várható el olyan kifinomult programozói tudás, amely grafikus programok megírásához szükséges.
- **Java**. A Java alapjában véve egy programozási nyelvet jelöl. Ezen a nyelven írták meg az EJS-t is. A Java nyelven írt programok sajátosságai [3]:
  - Bármilyen platformon (*Windows, Mac OS, Linux*, stb.) futtatható, azaz *platform független*.
  - Nyílt forráskódú (*open source*) termék. Mivel a forráskód szabadon elérhető, nem kerül pénzbe a program feltelepítése sem. Nem lehet tehát akadály az esetleges pénzsűke, a diákok is könnyen elérhetik interneten a terméket.
  - A világ egyik legelterjedtebb programozási nyelve, a többivel összehasonlítva egyszerű megtanulni, és jól használható programokat lehet vele írni.
  - Magas szinten támogatja az interneten való hozzáférést, így a Java nyelven készült alkalmazások nem csak az iskolából, hanem otthonról, interneten keresztül is elérhetők, jó hatékonysággal futtathatók.

Az EJS fejlesztői pont a fent felsorolt okokból választották a Java-t.

- **Simulations**, azaz szimulációk. Az EJS-be integrálva vannak a leghatékonyabb numerikus algoritmusok, amelyek közül csak ki kell választani a nekünk tetszőt. Egy szimuláció két jól elkülönülő részre bontható: a *fizikai modellre*, amelyben lényegében csak a mozgásegységek differenciális alakját kell megadni, és a *nézeti képre*, amelyben a modell változóit (hely, sebesség koordináták) grafikus objektumok paramétereire lehet kötni. (1. ábra).



1. ábra: Az EJS grafikus felülete.

Az EJS-ről a legtöbb információ a program „Wiki-oldaláról” érhető el [2]. Ezen az oldalon található meg az aktuálisan legfrissebb verzió is, amely ingyenesen letölthető. A program futtatásához csak egy Java alapú futtató környezetre van szükség (*Java Runtime Environment, JRE*), amelynek telepítője a legelterjedtebb operációs rendszerekre ingyenesen elérhető az alábbi URL-en: <http://java.sun.com/javase/downloads/index.jsp>.

## PÉLDÁK EJS-BEN KÉSZÜLT SZIMULÁCIÓKRA

### *LISSAJOUS-GÖRBÉK*

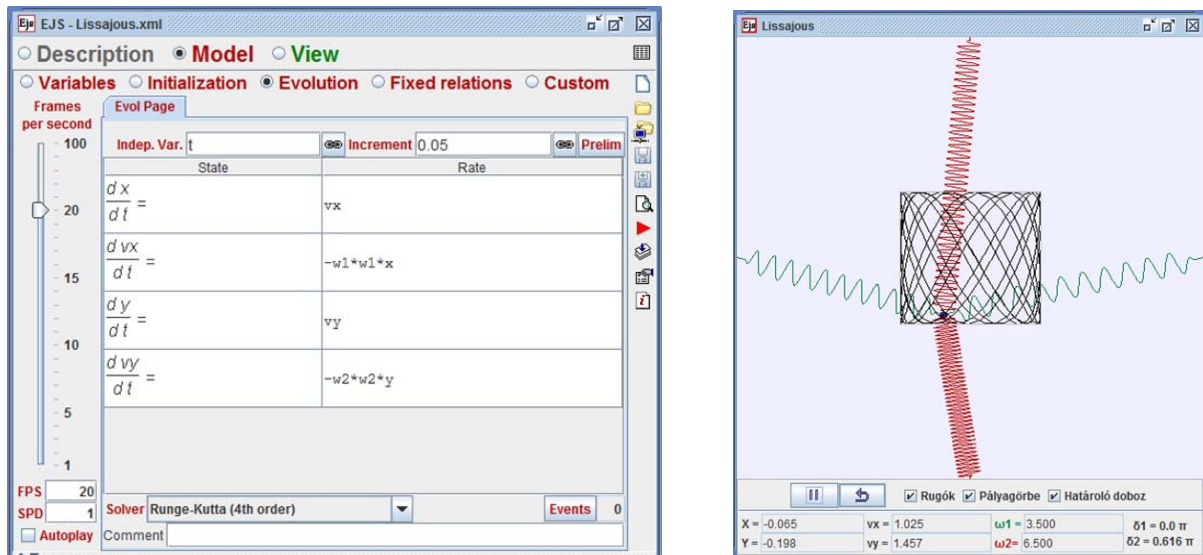
Az  $x$  és  $y$  irányban is harmonikus rezgést végző test mozgása az idő függvényében az

$$x(t) = A \cdot \cos(\omega_1 t) \quad , \quad y(t) = B \cdot \cos(\omega_2 t + \delta)$$

összefüggésekkel adható meg [5].  $A$  és  $B$  a rezgések amplitúdója,  $\omega_1$  és  $\omega_2$  a rezgések körfrekvenciája,  $\delta$  a fázisszög. A képletek alapján nehéz elképzelni, hogy a valóságban hogyan játszódik le ez a mozgás. A diákok túlnyomórészt a tanítási órán szembesülnek először merőleges összetételű rezgésekkel (Lissajous-görbékkel). Laboratóriumi körülmények között sem könnyű ilyen előállítani, nemhogy a valós életben megtapasztalni.

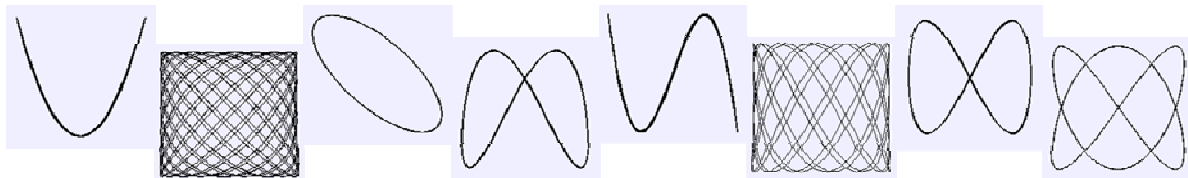
A jelenség megértéséhez az EJS keretrendszerben egy szimulációt készítettünk [1]. A program neve *Lissajous*, képernyőképe a 2. ábrán látható.

Ezen a felületen egy rugós kísérlet sematikus ábrája jelenik meg. Próbatestünket a szürke határoló dobozon belül tetszőleges pozícióba helyezhetjük, a mozgás ebből a helyzetből indul. Alul számszerűen is megjelennek a főbb paraméterek, amelyeket manuálisan is változtathatunk. Ha egy paramétert kézzel változtatunk meg, az ábra automatikusan az új értékeknek megfelelő állapotot mutatja. (Kézi beavatkozásra csak akkor van lehetőség, ha a szimuláció áll.)



2. ábra: Jobb oldalon a merőleges rezgések összetételét szimuláló program felülete, bal oldalon a szimuláció alapját képező mozgásegyenletek ( $\ddot{x} = -\omega_1^2 \cdot x$ ,  $\ddot{y} = -\omega_2^2 \cdot y$ ) szerkesztőablaka.

A mozgást bármikor leállíthatjuk és újra is indíthatjuk. *Checkbox*-ok segítségével az elemek láthatóságát vezérelhetjük (rugók, kirajzolt pályagörbe, határoló doboz). A program ellenőrzi a kezdeti feltételeket, figyel arra, hogy a dobozon túl ne kerülhessen a test. Ha bármit úgy akarnánk beállítani, hogy a létrejövő mozgás túlmenne a határoló dobozon, akkor a program a kezdeményezett változtatást nem hajtja végre. Erről egy hibaüzenettel értesíti a felhasználót.



3. ábra: A programmal előállított jellegzetes Lissajous-görbék.

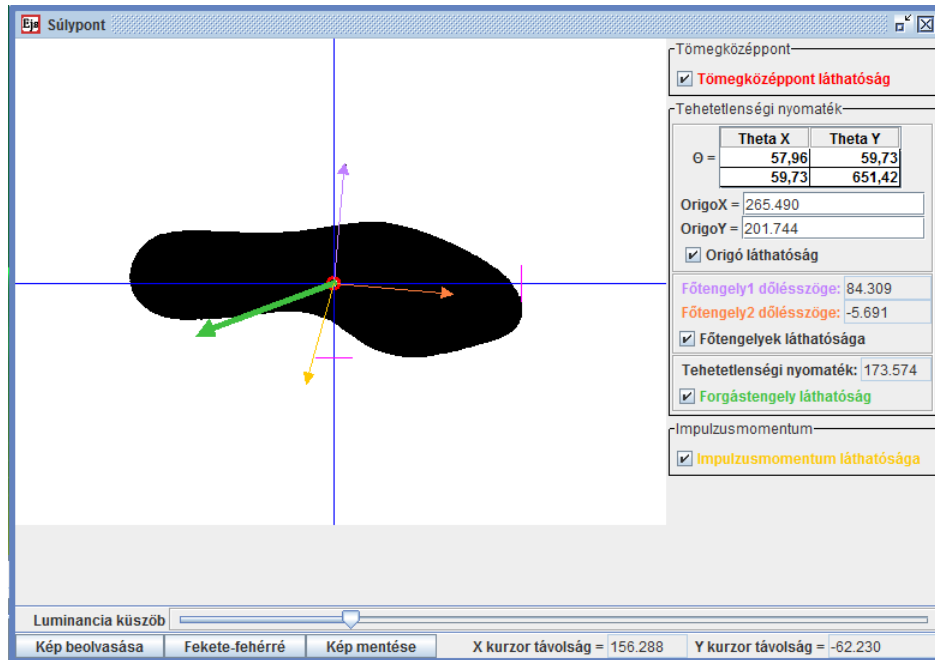
## TÖMEGKÖZÉPPONT ÉS TEHETETLENSÉGI NYOMATÉK

A *tömegközéppont* és *súlypont* a mechanika legfontosabb alapfogalmai közé tartozik. Pontrendszerknél és merev testeknél e fogalmak segítik a rendszer mozgásának, egyensúlyának vizsgálatát. (A nehézségi erőterben levő merev test egyensúlyát gyakran a súlypont helyzetével jellemezzük.) Ezért is fontos, hogy a diákok, akik a fizika óra keretein belül találkoznak ezekkel a fogalmakkal, jól értsék, érezzék ezek jelentését.

Merev testek forgó mozgásánál a szögsebesség mellett a *tehetetlenségi nyomaték* segítségével tudjuk megadni a test mozgási energiáját. Az említett fogalmak megértésének segítésére készítettük a „*Súlypont*” nevű programot, amelynek képernyőképe a 4. ábrán látható [1].

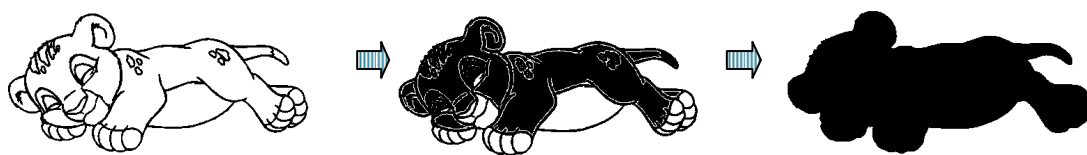
A „*Súlypont*” programmal síkbeli pontrendszereknek tömegközéppontját, tehetetlenségi nyomatékát vizsgálhatjuk. A lefényképezett alakzat JPEG, vagy PNG formátumban beolvasható, és fekete-fehérré alakítható. Az így kapott síkidomnak a mechanikában ismert összefüggésekkel [5] megadható a tömegközéppontja, tehetetlenségi nyomatéka, és más fontosabb jellemzői.

A kiválasztott képet a program a lehető legnagyobb méretben beilleszti a munkaterületre úgy, hogy a kép átméretezése közben a vízszintes és a függőleges oldalak aránya ne változzon meg. A felhasználó célja, hogy minden olyan képpont, amely a testhez tartozik, figyelembe legyen véve a megfelelő fizikai mennyiségek kiszámításánál. Ehhez szükségünk van arra, hogy a képet „értelmes módon” fekete-fehérré tudjuk alakítani.



4. ábra: A "Súlypont" program felülete

A program a beolvasott képpontok R, G és B komponenséből kiszámolja, hogy az adott képpont mennyire fényes, vagyis mennyi a luminanciája. Egy képpont attól függően lesz fehér, vagy fekete, hogy a fényessége az aktuális küszöbérték felett, vagy alatt található. Az alsó sávban látható „Luminancia küszöb” feliratú csúszkával lehet állítani ezt az értéket (5. ábra). A megfelelő eredményhez feltétlenül szükséges, hogy a bemeneti kép színei elegendően kontrasztosak legyenek!



5. ábra: Luminancia küszöb növelése

A fekete fehér képen a program a tömegközéppont definíciója alapján számolja a megfelelő koordinátákat, amelyet egy piros karikával jelenít meg (6. ábra). Ezen kívül a további fizikai jellemzőket is megmutatja (4. ábra):

- *Aktuális koordinátarendszer (kék vonalak):* amelyből a különböző fizikai mennyiségeket számoljuk. Az origó a kép tetszőleges pozíciójába áthelyezhető vagy az egérrel, vagy a koordináták megadásával az „OrigoX”, „OrigoY” mezőkben. A koordinátatengelyek külön-külön is mozgathatók vízszintesen és függőlegesen. A tengelyeken egy-egy lilás (magenta) színű kurzor látható, amelyek a tengely vonalának irányában mozgathatók. Ezeknek a kurzoroknak az origótól vett távolsága

megjelenik a program jobb alsó sarkában, így alkalmassá válnak arra, hogy a képről  $x$  és  $y$  irányú méreteket vehessünk le.

- *Főtengelyek iránya (halványlila és narancssárga nyíl):* A főtengelyek vízszintessel bezárt szöge a jobb oldali panelen számszerűleg is megjelenik, pontosabb mérések elvégzéséhez.
- *Pillanatnyi szögsebességvektor (vastag zöld nyíl):* Ennek a vektornak az iránya és a hossza is változtatható a zöld nyíl végének a mozgatásával. Mivel a szögsebesség mindig a tengely irányába mutat, ez a nyíl egyúttal a forgástengelyt is megadja.
- *Az impulzusmomentum-vektor nagysága és iránya (sárga nyíl).*
- *A tehetetlenségi nyomaték tenzora ( $\Theta$  mátrix) az aktuális koordináta-rendszerben felírva:* A mátrix elemei a koordináta-rendszer bármilyen áthelyezésekor újra kiértékelődnek.
- *A tehetetlenségi nyomaték („Tehetlenségi nyomaték mező”):* Adott forgástengely mellett egy kvadratikus alakkal számítható.

Figyelnünk kell arra, hogy ez a program nem egy munkaeszköz, hanem egy demonstrációs célú segédeszköz. A felhasználási gyakorisága általában 1-2 alkalom, ennyi idő alatt kell gyorsan, és hatásosan betöltenie a funkcióját. A tanár döntése, hogy hol húzza meg a határt a megjeleníthető információk széles skáláján.



6. ábra: A „Súlypont” programmal számolt tömegközéppontok jellegzetes síkidomoknál.

## IRODALOM

1. Gál Tamás Zoltán: Fizikai szimulációk megvalósításának egy eszköze: az EJS keretrendszer, szakdolgozat, ELTE, TTK, Budapest, 2009.
2. Easy Java Simulations Wiki, 2008. nov. 18., <http://www.um.es/fem/EjsWiki/index.php>
3. Sun Microsystems Inc.: About the Java Technology (The Java Tutorials), 2008. febr. 2., <http://java.sun.com/docs/books/tutorial/getStarted/intro/definition.html>
4. Open Source Physics, <http://www.compadre.org/osp/>
5. Tasnádi Péter, Skrapits Lajos, Bérces György: Mechanika I. , Dialóg Campus Kiadó, Budapest-Pécs, 2004.

## SZERZŐK

Gál Tamás Zoltán, mérnök informatikus, QSX Kft., Budapest, [gtz@ludens.elte.hu](mailto:gtz@ludens.elte.hu)  
Bérces György, egyetemi adjunktus, ELTE, TTK, Anyagfizikai Tanszék, Budapest, [berczes@metal.elte.hu](mailto:berczes@metal.elte.hu)