



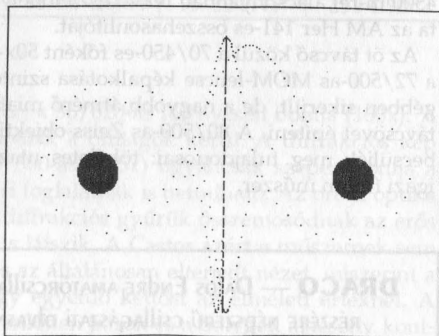
Számítástechnika

Gravitáció szimulátorok II.

Cikkünk első részében olyan programokról ejtettünk szót, melyek a gravitációs térben való mozgást egy nagy tömeg erőterében haladó elhanyagolható tömegű próbatestek pályái segítségével szemléltették. Ez az egyszerűsítő feltétel sok esetben kielégítő eredményt ad, ám nemegyszer megengedhetetlen hibákat is okoz. Lássuk tehát azon programok rövid bemutatását, melyek modelljében a testek tömegei egymáshoz képest nem elhanyagolhatók, tehát kölcsönösen befolyásolják egymás mozgását.

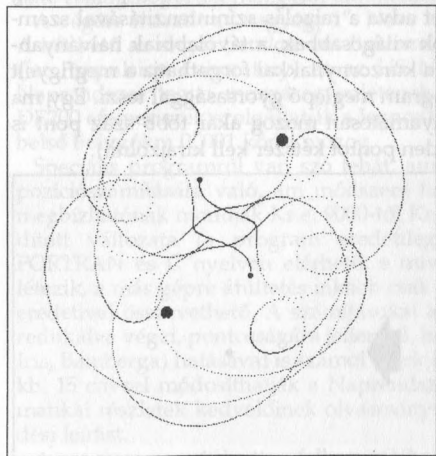
Melyek a legegyszerűbb elvárások, melyeknek a probléma ismeretében a modellező programnak eleget kell tenni? Vegyük szépen sorba. Most semmiképp sincs annyi próbatestre szükségünk, mint a már megismert galaxis-modellezésnél, hiszen például a Naprendszer fő mozgásainak modellezésére is elegendő tíz, vagy ha a nagyobb holdakat, kisbolygókat is be kívánjuk venni, néhányszor tíz test. Ezen testek jellemzőit viszont sokkal pontosabban kell tudnunk megadni, hiszen a végeredményt csak a kiinduló feltételek lehetőleg pontos ismeretében értékelhetjük, érthetjük meg. Melyek ezek a jellemzők? A tömeg, a térbeli hely és a szintén térbeli sebesség. El kell tudnunk helyezni tehát a térben maximum néhány tucat tömegpontot, kijelölnünk sebességük nagyságát és irányát. Mivel a szimulátorok továbbra is numerikus integrálással számítják a testek mozgását, illendő, hogy beleszólásunk legyen az integrálás lépésközébe. Ha mindezeket beállítottuk, még néhány billentyűleütés, egérkattintás, és székünkben hátradőlve szemlélhetjük a kezdődő színjátékot. Persze ehhez kell még valami fontos: a programnak nem árt, ha grafikus ábrázolási funkciója is van, azaz a testek mozgását mint egy rajzfilmet képes a monitoron megjeleníteni (bár az igazi az, ha kérésre a kiszámolt adatokat táblázatos formában is meg tudja mutatni).

A leírt elvárásokat több program is teljesíti. Nézzük, mit tudnak a valóságban! Az első a **GRAVITY 2.0**. Ez egy DOS-hoz készült program, melynek kezelése menürendszerén keresztül szinte magától értetődő. Egyszerre maximum 16 testet tud kezelni, a tapasztalatok szerint ez általában elegendő. A testek definiálására a „Planet Editor” menüpont szolgál, ahol leírhatjuk a fent felsorolt jellemzőket. A tömeg, helyzeten és sebességeken kívül minden „planétára” megadhatjuk az átlagsűrűségét is, ebből az adatból számítja a gép az égitest gömbjének



1. ábra

méretét. Ha kívánjuk, itt jelölhetjük, hogy a test mozgása során rajzolja-e a pályagörbét, esetleg erőszakkal rögzíthetjük a tér egy megadott pontjához (innen a szimuláció alatt nem mozdulhat el, bár erőtere hat a többi testre). Ez utóbbinak a „valós” helyzetek vizsgálatánál nincs helye, érdekességként kipróbálható. A számítást a „Go” menüponttal indíthatjuk, az bármikor megállítható és továbbindítható. Az „Examples” menüpont néhány előre definiált adatsorral tölti fel a testek táblázatát. Tanulmányozhatjuk itt a Hold Föld körüli keringését, kettős vagy többszörös csillagok pályáját vagy akár egy naprendszer mozgásait. A programmal való ismerkedést célszerű ezekkel a példákkal kezdeni. A program „Adjust constants” menüpontjában adhatjuk meg a szimuláció lépésközét, s ha úgy tartja kedvünk, a gravitációs konstans értékét is megváltoztathatjuk. A program a legelterjedtebb grafikus kártyákat támogatja.



2. ábra

Az 1. ábra két, igen nagy tömegű test gravitációs erőterében mozgó harmadik tömeg pályáját mutatja, míg a 2. ábrán szintén két nagyobb tömeg körül keringő másik két kisebb test mozgása látható.

Szintén **GRAVITY** a neve a TMA programjának. Nemcsak a név, a program is hasonló, bár ez utóbbi Windows-hoz készült. Szinte egyetlen nem túl lényeges különbség, hogy itt a „bolygók” helyeit és sebességét az egér nyilacská-jával jelölhetjük ki. A szimuláció a jól ismert Windows ablakban folyik. Egy harmadik DOS-os, ám 16 bites védett módban futó program az **NBODY16**. Eszközeiben kissé szegényesnek tűnő, de gyors szimulációs program, melyben a testek a Windows-oshoz hasonlóan egérrel megadandók. A kitűzött célt

végző soron mindhárom program teljesíti: szép, grafikus ábrákon szemléltetik az egymás erőterében haladó testek mozgását.

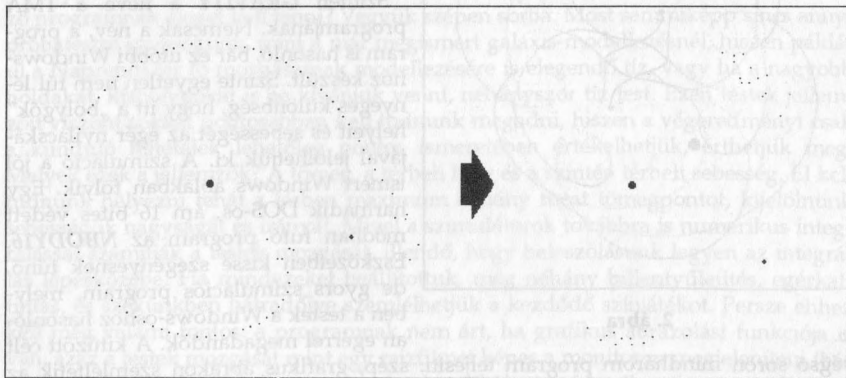
Az előbbi programok közös hiányossága, hogy a modellezés nem valódi három dimenzióban (3D) folyik, csupán síkban. Ha a valós rendszereket tekintjük (pl. Naprendszer), ez a fogyatékoság, bár bosszantó, nem teszi használhatatlanná őket, hiszen sokszor az égitestek mozgása valamilyen síkhoz közeli pályákon folyik...

Az elmondottak nyomán jogosan alakul ki a Tisztelt Olvasóban a kép, hogy nincs program, amely tisztességesen, mindenre kiterjedő gondossággal, valódi 3D-ban végeznél el feladatát? De igen, van! Lássunk erre is két példát.

Hogy, hogy nem, Jens Jorgen Nielsen programját szintén a **GRAVITY** névvel illette. Valóban profi munkával állunk szemben, olyan tudással és teljesítménnyel, melyet nehéz feltételezni az alig 46 kB méretű kódról. A Windows-on elkényelmesedett felhasználó kicsit fanyalogat, de logikáját megértve a program kezelése egyszerű. Az összes bemenő adatot ugyanis egy leíró állományban kéri, melyet a parancssorban adhatunk át később neki. A csomagban néhány példát is találunk (alapértelmezett kiterjesztésük .BDY). A leíró adatok első sora a szimulációs paramétereket állítja be.

Itt szerepel sorban az idő lépésköze, a gravitációs konstans, a képernyőmód (később részletezendő), a megfigyelő távolsága a testektől és egy „finomító” tényező, mely a numerikus integrálásból fakadó hibák egy részét javíthatja szükség esetén. A többi sor a testeket írja le. A már tárgyalt tömeg-, pozíció- és sebességadatokon kívül itt adható meg a test megjelenítésének színe és formája is.

Ha mindezt begépeztük, indíthatjuk a programot és ha van véletlenül egy piroszöld szemüvegünk, már láthatjuk is a csodát: a monitor megelevenedik, a mozgó pontok kiemelkednek a képernyő síkjából, kavarnak, keringenek olyan látványt nyújtva, melyet eddig még egyetlen másik programnál sem élhettünk át. Igen, itt nem csupán a szimuláció, de a megjelenítés is három dimenziós, igaz, a látvány élvezéséhez speciális szemüveg szükséges. Akinek ilyen nincs, használhatja a másik két képernyőmódot. Az egyik a definíciós leírásban megadott színekben jeleníti meg a tömegpontokat, a másik némi térélményt adva a rajzolás színintenzitásával szemlélteti a távolságot (a közelebb lévő pontok világosabbak, a távolabbiak halványabbak). Akinek pedig még ez sem elegendő, a kurzornyilakkal forgathatja a megfigyelt térrészt tetszőleges irányba. Mindezt a program meglepő gyorsasággal teszi. Egy ma már nem túl ritka 486DX/33-as gépen folyamatosan mozog akár több száz pont is (ne feledjük, a térbeli megjelenítéshez minden pontot kétszer kell kirajzolni!).



3. ábra

A látvány tehát elsősorú, de a program szimulációs képességei sem maradnak el mögötte. Vizsgáljuk például a háromtest problémát: helyezzünk el egy nagyon nagy tömegű égitestet a képernyő közepén (Nap) és állítsunk körülötte körpályára egy szintén nagy, de az elsőnél jelentősen kisebb tömeget (Jupiter). Ezután a kisebb égitest pályája mentén egyenesen helyezünk még el vagy száz igen kicsi, de nem elhanyagolható tömegű égitestet (kisbolygók), és lássuk mi történik! Kezdetben igen nagy a kavargás, a rend felbomlik, a Jupiter kisöpri környezetéből a vele azonos pályán haladó kisbolygókat, a kis testcskék jó része kirepül a rendszerből, néhányuk befelé lökődik és elnyújtott keringést kezd a Nap körül, ha szerencsénk van, a Jupiter is befog egy-kettőt közülük. Nem is várunk mást, hiszen törvényszerű: az óriásbolygó perturbáló hatása miatt nem maradhat égitest vele azonos pályán. Ha azonban kicsit tovább várunk, furcsa dolgot láthatunk. A kisbolygók két csoportja mintha játszana a Jupiterrel: egyik részük a bolygó előtt halad, mintha menekülne, a

másik csoport pedig követi azt. Mindkét felhőske kavarog, de időben stabilan halad a Jupitertől körülbelül azonos távolságban, azzal nagyjából megegyező pályán (3. ábra). Élesebb szeműeknek az is feltűnik, hogy a Napból nézve a „menekülő” és a „követő” csoport közel azonos szögtávolságban, 60° -ra van a Jupitertől. Igen, szimulációkkal megtaláltuk a Nap-Jupiter rendszer L_4 és L_5 librációs pontjait (Lagrange-pontok), melyek közelében (a pontok körül hosszú periódusú rezgést végezve) harmadik test (egy testek) keringése lehetséges. Ha már számításunkban nevet adtunk a központi égitestnek és a nagybolygónak, megadhatjuk a két kisbolygófelhő naprendszerbeli nevét, hiszen a valóságban is léteznek: ezek képezik a trójai kisbolygók csoportját.

Hasonlóan érdekes eredményeket kapunk, ha egy galaxis-ütközést vagy kölcsönható csillagcsoportok mozgását tanulmányozzuk a program segítségével.

A végére maradt a „nagyágyú”. Neve nem túl sokatmondó *DE118* vagy *DE200*. A két betű az angol *Development Ephemeris* rövidítése. A programot a kaliforniai JPL (Jet Propulsion Laboratory) és az US Naval Observatory készítette speciálisan a Naprendszer tagjai mozgásegyenletének numerikus integrálására. Jelenleg a JPL DE200 eferemiszek szolgáltatják a legpontosabb bolygókoordinátákat. Pontosságuk a belső bolygókra $0,001$ körül mozog.

Speciális programról van szó tehát, amely kizárólag a Naprendszer égitestjeinek pozíciószámítására való, ám módszere hasonló az eddig ismertettekhez. Adatait megbízhatónak mondják Kr.e. 9000-tól Kr.u. 13000-ig. A PC kategóriájú gépekre fordított változata (a program eredetileg „nagygépre” íródott, de a forráskód FORTRAN és C nyelven elérhető, s mivel C fordító szinte minden számítógépre létezik, a más gépre áttöltés inkább csak idő és ügyesség kérdése) pontosságban az eredetivel összevethető. A számításokat a program a rendszer tömegközéppontjára redukálva végzi, pontosságára jellemző, hogy még öt kisbolygó (Ceres, Pallas, Vesta, Iris, Bamberga) hatásával is számol (ezek összesen 10^{-12} csillagászati egységgel, azaz kb. 15 cm-rel) módosíthatják a Naprendszer tömegközéppontjának helyét). A matematikai részletek kedvelőinek olvasmányul ajánlom a programhoz mellékelt működési leírát.

A program nem túlzottan felhasználóbarát. Működéséhez a futtatható kódon kívül egy definíciós állomány is szükséges, ebben található a Naprendszer égitestjeinek baricentrikus koordinátái JD 2 440 400,5-re. Ezek a kiinduló adatok, melyekből integrálással az összes többi szolgáltatja a program. Elindítás után még néhány adatot be kell billentyűznünk: a számítás lépésközét, a kívánt adatok számát és néhány segédváltozt, melyek a pontosságot befolyásolják. Ezek után már csak jó adag türelemre van szükségünk, hogy az eredményt kivárhassuk. Minél távolabb van a kívánt időpont a kiindulástól, annál több idő szükséges a számításhoz (l. a sorozat első cikkét: numerikus integrálás). Ha minden rendben ment, a végeredmény még nem igazán látványos: egy akár több tíz MB-os állomány, mely a kért időpontokra a bolygók és a Hold valóban precíz adatait tartalmazza, „természetesen” csak a számítógép számára értelmes, bináris formában. Az adatszerkezet ismeretében mindenki maga írhatja meg a programot, mellyel ebből az adatszúngelből a számára fontosakat kizmazsolazza és szemének tetsző formában megjeleníti. Szép feladat!

A program futtatását jó szívvel csak Pentium processzoros (osztási hibától mentes) masinával és tengernyi idővel rendelkező olvasóinknak ajánljuk, akik az elkövetkező hetekben nem kívánják gépüket más feladattal zaklatni...

Érdeklődők a két cikkben ismertett programokról másolatot kaphatnak mágneslemez és felbélyezett válaszborték ellenében a rovatvezetőtől.

HEITLER GÁBOR