

Számítógépes szimulációk

V.: Populációdinamika

Pál Balázs*

*Eötvös Loránd Tudományegyetem

2019. április 27.

Abstract

A *Számítógépes szimulációk* laboratórium ötödik alkalmával egy, a fizika területén kívül eső témával foglalkoztunk, mely inkább a biológia tudományának részét képezi. Ez a téma a populációdinamika volt, mely az egyes állat-, esetlegesen emberpopulációk időbeli változásával, valamint a környezettel és más fajokkal való kölcsönhatásával foglalkozik. A téma azért szerzett létjogosultságot magának a tárgy anyagába történő bekerüléséhez, mert az ilyesfajta időbeli változásokat legjobban numerikus differenciálegyenlet megoldásokkal modellezhetjük. A feladatok során megvizsgáltuk a logisztikus-, a csatolt-logisztikus-, valamint a Lotka–Volterra-modellt is, melyek a legalapvetőbb populációdinamikai algoritmusoknak számítanak.

1. BEVEZETŐ

A labor ötödik gyakorlatán a populációdinamika témakörével foglalkoztunk. A téma kitűzésének motivációja, hogy bemutassa, a fizikán kívül más témakörökben is hasznos (és egyáltalán lehetséges) a differenciálegyenletek numerikus megoldásával foglalkozni. Fontos természetesen megemlíteni, hogy az ezekből kapott modellek és adatok csak a valóság egy közelítésének felelnek meg, mely leginkább az összefüggések és markánsabb karakterisztikák elemzésére alkalmasak. Ennek oka, hogy az ilyen események erősen sztochasztikus, erősen nemlineáris jellegűek, így az itt tárgyalt megközelítések nem alkalmazhatóak pontos vizsgálatokra. Azonban céljuk nem is ez, hanem az előbb már említettek: a lényegi motívumok feltérképezése.

A használt modellek - ahogy a gyakorlat leírása is fogalmaz - egy általános keretrendszernek tekinthetők, melybe nem csak a populációdinamika konkrét problémái, hanem teljesen másfajta jelenségek is beilleszthetők. Ilyen esetekben a hasonló alakú egyenletek, szintén hasonló karakterisztikájú dinamikát jeleznek előre, legyen szó akár kémiai, akár gazdasági problémáról[1]. A jelenlegi jegyzőkönyvben egy ilyen keretrendszer megteremtéséhez használható modelleket fogunk vizsgálni.

2. FELADATOK

A gyakorlat során az eddigiekben minden feladat mellé egy, az alapokat tartalmazó példakódot kapunk. Ez az első téma, ahol ilyen nem áll rendelkezésünkre, így minden szükséges forráskódot magunknak kell megírunk. Természetesen az eddigi kódok használata nem tiltott, ezeket fel is használtam a szimuláció programkódjának megírásához.

A feladatok során a populációdinamikában is ismert logisztikus-, csatolt-logisztikus-, valamint Lotka–Volterra-modelleket kellett vizsgálnunk.

Ezek közül az első említettel kapcsolatban a fixpontok körüli viselkedést kellett elemeznünk, pon-

tosabban, hogy mennyire adja vissza az analitikus eredményt a fixponttól vett távolság függvényében. A kérdéses viselkedést mind az Euler-, mind pedig a Runge–Kutta-módszerek esetében meg kellett vizsgálnunk. Végül a gyakorlat leírásában található ábrákat kellett megfelelő paraméterek megválasztásával reprodukálnunk.

A csatolt-logisztikus-modell esetében be kellett bizonyítanunk a *kompetitív kizárás* elvét, mely kimondja, hogy $\alpha = \beta = 1$ paraméterű modell esetén két, egymással versengő állatfaj nem létezhet egymás mellett egyensúlyban, a nagyobb k paraméterű¹ faj kiszorítja a másikat az élőhelyről. Meg kellett azt is mutassuk, hogy egyensúly csak azonban az esetekben lehetséges, ahol teljesülnek az $\alpha k_2 < k_1$ és $\beta k_1 < k_2$ feltételek.

Végül az utolsó modellt, a Lotka–Volterra-t kellett implementálnunk, majd annak fázisterét ábrázoljuk. Ezt a véges táplálék és telítődés hatásainak figyelembe vételével, valamint anélkül is meg kellett valósítsuk.

3. ELMÉLETI ALAPOK

3.1. ALKALAMAZOTT MODELLEK

Ebben a jegyzőkönyvben három populációdinamikai modellt vizsgáltunk. Ezek sorrendben a logisztikus-, a csatolt-logisztikus-, valamint a Lotka–Volterra-modellek voltak. Ezeket az alábbiakban a gyakorlat leírásának logikáját követve vázoló részletekben.

3.1.1. Logisztikus modell

Vizsgáljuk egy állatpopuláció gyarapodását az idő függvényében. Olyan esetben, amikor nincsen semmilyen befolyásoló tényező, mely vagy negatívan érintené, vagy korlátokat szabna a populáció fejlődésének, a növekedés sebessége az egyedszámmal lesz arányos. A populáció létszámának időfejlődése

¹ k jelentése a (3.1.1)-es alfejezetben tisztázott

ekkor könnyen kifejezhető az alábbi módon:

$$n(t + \Delta t) = n(t) + w_{be}n(t) \quad (1)$$

ahol w_{be} az ún. *szaporodási ráta*. Ez a ráta határozza meg, hogy Δt idő alatt hány utód születik a populáció létszámához viszonyítva. Triviális, hogy minél nagyobb w_{be} , annál gyorsabb a növekedés és fordítva. Ez alapján felírhatjuk a rendszer Master-egyenletét is, opcionálisan folytonos közelítésben:

$$\frac{dn(t)}{dt} = w_{be}n(t) \quad (2)$$

Ez a folytonos határeset akkor pontos, ha a populáció létszáma nagy, és emiatt az w_{be} ráta folytonossá tétele nem okoz gondot.

Az egyszerű (2)-es differenciálegyenlet végleges megoldása ismert, ez egy exponenciális függvény:

$$n(t) = e^{w_{be}t} \quad (3)$$

Tehát a faj egyedszáma folyamatosan nőni fog a végtelenségig, a (3)-as egyenletnek megfelelően. Hogy a valóságos helyzetet jobban tükrözzük, vezessünk be egy ún. *halálozási rátát* is az eddigi w_{be} mellé, és nevezzük ezt el w_{ki} -nak. Ekkor a Master-egyenlet a következőképpen alakul:

$$\frac{dn(t)}{dt} = -w_{ki}n(t) + w_{be}n(t) \quad (4)$$

Triviálisan látható, hogy az $-w_{ki} + w_{be}$ mennyiség előjelétől függően a populáció folyamatosan nő-e, vagy csökken. A (4)-es differenciálegyenlet megoldása az előbbihez nagyon hasonló:

$$n(t) = e^{(w_{be} - w_{ki})t} \quad (5)$$

Egy újabb szorzótényezővel, mely n egyedszám függvénye, vegyük figyelembe, hogy a faj számára rendelkezésre táplálék, vagy élőhely korlátos mennyiségű/méretű:

$$\frac{dn(t)}{dt} = (-w_{ki} + w_{be}) \cdot n(t) \cdot F(n) \quad (6)$$

Vegyük $F(n)$ legegyszerűbb, lineáris modellét, mely kielégíthető az alábbi függvénnyel:

$$F(n) = 1 - \frac{n(t)}{k} \quad (7)$$

Ekkor, ha $n = k$, a populáció nem nőhet tovább, ezzel egy felső határt szabva a faj egyedszámának. A (6)-os egyenletet ezzel kibővítve az alábbi formát kapjuk:

$$\frac{dn(t)}{dt} = (-w_{ki} + w_{be}) \cdot n(t) \cdot \left(1 - \frac{n(t)}{k}\right) \quad (8)$$

Ennek a differenciálegyenletnek a megoldását már könnyen felírhatjuk a következő módon:

$$n(t) = \frac{k \cdot n_0}{n_0 + (k - n_0) \cdot e^{-(w_{ki} + w_{be})t}} \quad (9)$$

Ezt a kapott eredményt hívjuk logisztikus egyenletnek, melynek alakja egy növekedő, vagy csökkenő, szigmoid-jellegű görbe általános esetben. Mikor $-w_{ki} + w_{be} = 0$, a kapott görbe alakja egy konstans egyenes.

A numerikus vizsgálat esetében majd a (8)-as egyenletet fogjuk megoldani, míg az analitikus megoldás ábrázolásánál a (9)-es egyenletet fogjuk illesztrálni.

3.1.2. Csatolt-logisztikus-modell

Vizsgáljunk azt az esetet, amikor két faj él egyszerre egy területen, de egyik sem ragadozója a másiknak. Ilyen helyzetben ez a két populáció egymással fog versengeni a táplálékért és az élőhelyért. Az élelmesebb és életrevalóbb faj növekedik és túlél, a gyengébb pedig akár ki is halhat. Ezt az eddig tanultaknak megfelelően akár ki is következtethetjük, hogy ezt az esetet két, csatolt differenciálegyenlet írja le a fentiek alapján a következő módon:

$$\frac{dn_1(t)}{dt} = (-w_{ki_1} + w_{be_1}) \cdot n_1(t) \cdot \left(1 - \frac{n_1 + \alpha n_2}{k_1}\right) \quad (10)$$

$$\frac{dn_2(t)}{dt} = (-w_{ki_2} + w_{be_2}) \cdot n_2(t) \cdot \left(1 - \frac{n_2 + \beta n_1}{k_2}\right) \quad (11)$$

3.1.3. Lotka–Volterra-modell

Ez a modell az olyan esetek tanulmányozására alkalmas, amikor egy növényevő, és azon a növényevőn ragadozó faj kölcsönhatását szeretnénk szimulálni. Vegyük példának a téma bemutatásaiban klasszikusan használt nyulak és rókák kölcsönhatásának esetét. A nyulak növényevők, az élőhelyen csak növényeket esznek, és szaporodnak. A rókák szintén ezen az élőhelyen élnek, azonban nyulakkal táplálkoznak, és mellette szaporodnak. Ha elegendő számú nyúl van folyamatosan, a rókák száma elkezd növekedni, azonban ezzel együtt a nyúlfogyasztási rátájuk ki. Emiatt a nyulak populációjának növekedése elkezd lassulni, majd egy kritikus ponton negatívba, populációcsökkenésbe megy át. Ahogy a nyulak száma ezután csökken, a rókák tápláléka is. Emiatt a rókák egyedszámának növekedése elkezd lassulni, majd szintén negatívba fordul. Ahogy csökken a rókák és így egyben a nyulak ragadozóinak száma, a nyulak populációja újra növekedésnek indulhat. A rókák tápláléka ahogy növekedik, úgy a populációcsökkenésük rátája lecsökken, majd újra pozitívba fordul át. Ezután pedig kezdődik az egész előlről. Ezt a dinamikát próbálja a Lotka–Volterra-modell a valósághoz minél hűbben szimulálni.

Első megközelítésben éljünk azzal a feltételezéssel, hogy a nyulak (N_y) és rókák (R) fogyasztása sehogys sem korlátozott. Ez annyit jelent, hogy feltesszük, hogy a növények sosem fogynak el és mindig rendelkezésre állnak, a rókák nyúlfogyasztási képessége és éhsége pedig szintén konstans. Ezt a helyzetet az alábbi Master-egyenletekkel írhatjuk le, ahol a gyakorlat leírásában is használt jelölésmódot alkalmazzuk (R - F és N_y - R cseréjével):

$$\frac{dn_{N_y}}{dt} = a \cdot n_{N_y} - b \cdot n_{N_y} \cdot n_R \quad (12)$$

$$\frac{dn_R}{dt} = c \cdot n_{N_y} \cdot n_R - d \cdot n_R \quad (13)$$

Ahol a a nyulak, $c \cdot n_{\text{Ny}}$ pedig a rókák születésének, míg $b \cdot n_{\text{R}}$ a nyulak, d pedig a rókák halálozásának rátáját jelölik. Egyértelműen látszik az egyenletek alapján, hogy a nyulak halálozásának a rátáját a rókák száma, a rókák születési rátáját pedig a nyulak száma növeli egy szorzótényezőként manifesztálódva. A modellt realisztikusabbá tehetjük, ha a csatolt-logisztikus modellhez hasonlóan korlátozzuk a nyulak egyedszámát, azt feltételezve, hogy a rendelkezésükre álló táplálék nem véges (ahogy a valóságban is történik). Emellett adhatunk egy felső kapacitást a rókák nyúl fogyasztási képességeire is, hisz a valóságban se azonos ütemben, csillapítatlan éhséggel fogyasztják a nyulakat. Ezt az alábbi paramétercserékkel érhetjük el:

$$a \rightarrow a \cdot \left(1 - \frac{n_{\text{Ny}}}{k}\right) \quad (14)$$

$$n_{\text{Ny}} \cdot n_{\text{R}} \rightarrow \frac{n_{\text{Ny}} \cdot n_{\text{R}}}{1 + n_{\text{Ny}} \cdot s} \quad (15)$$

Bevezetve ezeket a változtatásokat a (12)-as és (13)-es egyenletek a következőképp alakulnak:

$$\frac{dn_{\text{Ny}}}{dt} = a \cdot \left(1 - \frac{n_{\text{Ny}}}{k}\right) \cdot n_{\text{Ny}} - b \cdot \frac{n_{\text{Ny}} \cdot n_{\text{R}}}{1 + n_{\text{Ny}} \cdot s} \quad (16)$$

$$\frac{dn_{\text{R}}}{dt} = c \cdot \frac{n_{\text{Ny}} \cdot n_{\text{R}}}{1 + n_{\text{Ny}} \cdot s} - d \cdot n_{\text{R}} \quad (17)$$

Melyből visszakaphatjuk az eredeti (12)-as és (13)-es egyenleteket a $k \rightarrow \infty$, $s \rightarrow 0$ határesetben. Épp emiatt csak az utóbbi, módosított egyenletet fogom használni a szimulációban, melyet megfelelően paraméterezve, közelíteni tudom hibahatáron belül akár a határesetet is.

4. MEGVALÓSÍTÁS

Az eddigiekkel ellentétben nem volt előzetesen egy forráskód sem megadva számunkra, így mindent nekünk kellett megírunk a feladatok teljesítéséhez. A forráskódokat C++ nyelven implementáltam, amihez felhasználtam az eddigiekben már többször is alkalmazott `cpl` library-t is. Ez azért is volt hasznos, mert abban rengeteg olyan, már eredetileg, vagy az eddigiek során megírt függvény is szerepel, amiket ebben a témában is alkalmaznunk kell. Emellett így meg tudtam valamennyire őrizni a kurzuson készített kódok integritását, tehát a külsőleg is azonos látvány hatását. Az egyes modellek számára 1-1 külön forráskódot hoztam létre, melyek egymástól teljesen függetlenek. Több szempontból ez a megközelítés nem feltétlen optimális, ugyanis az egyes forráskódok között csak nagyon kis különbségek találhatók, azonban így sokkal átláthatóbb és egyszerűbb formában tudtam őket egyesével megírni, így összességében javítva a programokon történő munka hatékonyságán. A forráskódokat az eddigiekhez hasonlóan egy saját batch file segítségével, benne a `clang` fordító felhasználásával fordítottam. A futtatható `exe` programokat egy Jupyter Notebook-ban futó Python 3 kernel segítségével indítottam, a szimuláció a kezdőfeltételeket szintén ebből a környezetből várja, bemenő paraméterek formájában.

A kimenet minden esetben egy `.dat` file, mely minden sora egy-egy szimulációs lépésnek felel meg. Ez

a file minden esetben tartalmazza a szimulált populációk aktuális számát, míg az adaptív lépéshosszal operáló esetben az aktuális lépésközöket is jegyzi. Emelett mind a fix-, mind az adaptív lépésközzel működő esetben a futásidőt is tartalmazza. A végleges forráskódok és a programokat futtató Notebook file mind elérhető GitHub-on[2].

5. KIÉRTÉKEELÉS

Ebben a témában az eddigiekhez képest jóval kevesebb feladatot kellett teljesítenünk, legalábbis összességében jóval könnyebbeket. Első lépésben miután elkészültek a működőképes programok, ábrázoltam mindegyik szimuláció által jósolt populáció időfejlődését, melyeket az (1) - (4) ábrákon demonstráltam. Ezeket sorrendben a logisztikus-, csatolt-logisztikus- és Lotka-Volterra-modellek által jósolt időfejlődések láthatóak. Az első kettő esetében több különböző w_{be} paraméter által generált görbesereget ábrázoltam, minden futtatásban azonos paramétereket használva, míg a Lotka-Volterra-modell-ről egy konkrét futtatás eredményét is kiemeltem a (3)-as ábrán.

Első feladatként meg kellett vizsgálnunk, hogy a logisztikus modell a fixpontja ($r = -w_{ki} + w_{be} = 0$) körül hogy viselkedik, és így a gyakorlat leírásában is látható ábrákat kellett reprodukálnunk. Ezeket 12 esetre megnézve a (5)-ös ábrán meg is tettem, a leírásban is használt paraméterekhez hasonló változókat használva. Így vissza is kaptam tökéletesen ugyanazokat az ábrákat, amiket ott is láthatunk. Ehhez a Runge-Kutta-Cash-Karp iteratív algoritmust használtam, azonban a feladat felhívására megvizsgáltam az esetet az Euler- és a negyedrendű Runge-Kutta-módszerrel is, immár csak 6-6 különböző r mellett. Ezeket a (6)-os, a kapott értékek különbségét pedig a (7)-es ábrán illusztráltam. Következő feladatban össze kellett hasonlítanunk a logisztikus modell elméleti, valamint a numerikusan kapott megoldását, majd ábrázolnunk kellett a különbségüket. Ezt két grafikonon ábrázoltam, fixpontos léptetési módszert alkalmazva. A kapott eredmények és a hozzájuk tartozó magyarázat a (8)-as ábrán látható.

A következő feladat a csatolt-logisztikus-modell vizsgálatára vonatkozott, mely a stabilitás kérdését járta körül. Azzal a kezdeti feltételezéssel indultunk, miszerint egy $\alpha = \beta = 1$ paraméterekkel rendelkező modell sosem lehet stabil, kizárólag a $k_1 = k_2$ esetben. Minden más érték mellett a nagyobb k paraméterű faj győzedelmeskedik hosszú távon, teljesen kipusztulásra ítélve a másik populációt. A (9)-es ábra valamelyest betekintést nyújt a kérdés megválaszolásába. Az ábrák magyarázata szintén a képaláírásban olvasható részletesen. A csatolt-logisztikus-modell fázisterét két paraméter változtatásával is ábrázoltam, ezek a (10)-es és (11)-es ábrákon szerepelnek. Mindkét esetben mind a fix, mind pedig az adaptív lépéshosszváltó módszerekkel kapott eredményt ábrázoltam. A (10)-es ábrán a w_{be_1} születési ráta változtatásával kapott görbesereget demonstráltam, míg a (11)-es ábrán az előbb tárgyalt stabilitási problémához tartozót, melyet a k_1 maximális egyedszám változtatásával generáltam. Az utóbbi esetében nagyon szépen látszik az ábrán a stacionárius helyzet: ez a görbesereg

„két szárnya” közötti lineáris egyenes, mely két részre osztja a grafikont.

Utolsó feladatként a Lotka–Volterra-moddal kellett foglalkoznunk. Ennek fázistereit kellett ábrázolnunk, mind a csillapítás mentes, mind pedig a csillapított megvalósításai esetén. A 12-es ábrán a csillapítatlan eset $n_{Ny} - n_R$ fázistere, a 13 ábrán pedig a csillapított esté látható. Ezekben az a paramétert változtatva hoztam létre különböző görbéket. A paraméter változtatásának hatása igazán a második, csillapított esetben látszik ugyanis ez azt okozta, hogy a rendszer egyre gyorsabban és gyorsabban spirálzott be a fix, stacionárius állapotába. A több fajt is figyelembe venni képes programkód elemzésére már nem volt sajnos időm, azonban egy optimális megoldásával a későbbiekben majd megpróbálkozom, melyet a gyakorlat GitHub könyvtárában elérhetővé teszek.

6. FUTÁSIDŐ

A feladatok elvégzése után megpróbáltam megvizsgálni az algoritmusok futásidejét, azonban erre az

eddigiekkel ellentétben erősen kontraintuitívnak nevezhető eredményeket kaptam. Függetlenül attól, hogy mekkora időhosszra vizsgáltam a szimulációkat, a futásidő minden esetben nagyjából azonos volt, apró fluktuációk mellett. Erre próbáltam valami elfogadható választ találni, de eddig még nem sikerült.

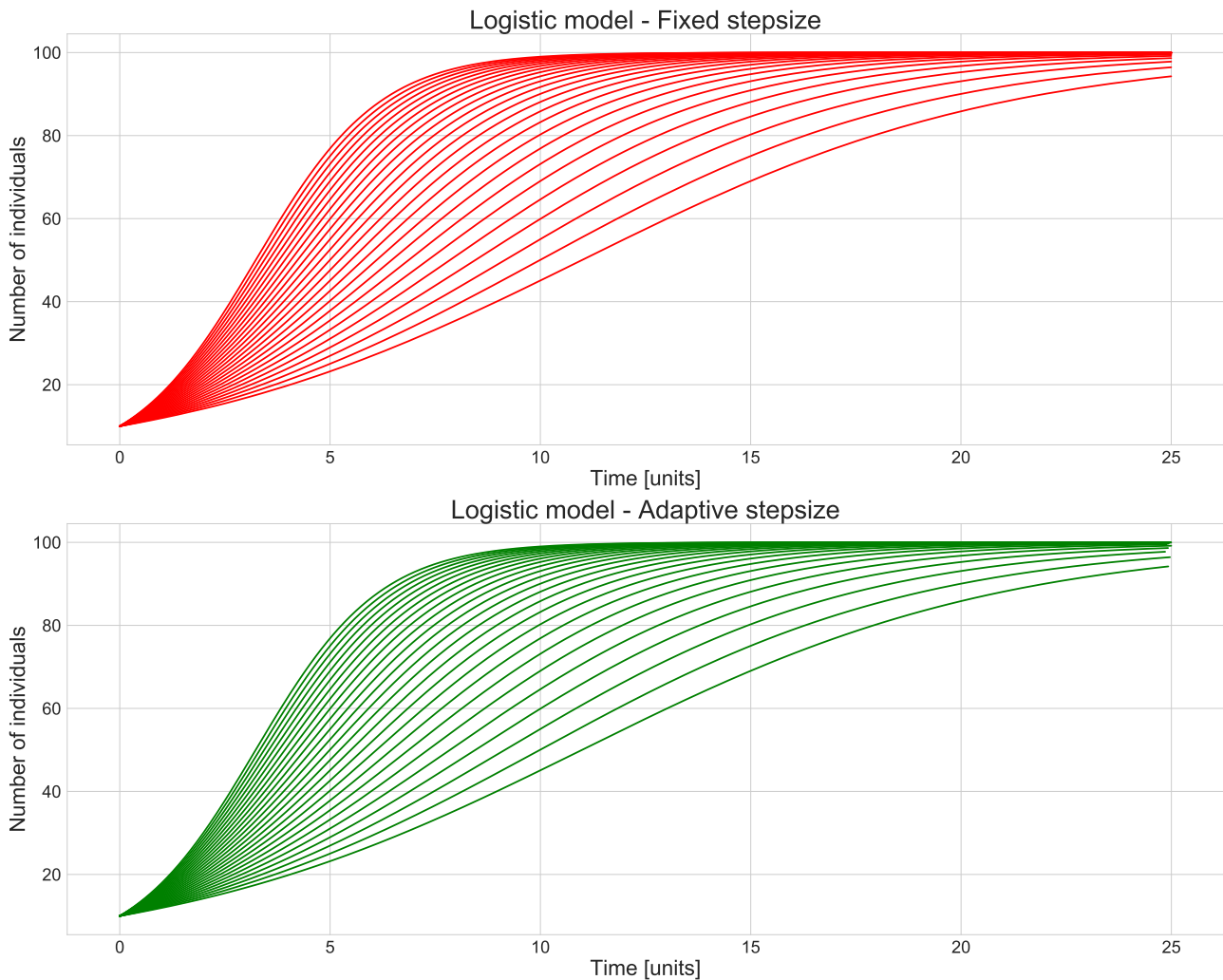
7. DISZKUSSZIÓ

A laborgyakorlat során megismerkedtem több újabb, a fizikán kívül is széleskörben használt algoritmussal, a logisztikus, valamint a Lotka–Volterra-moddal. Megvizsgáltam, hogy ezek a populációdinamika témakörében hogyan teljesítenek és az elméletekkel megegyező, a valósággal is összecsengő karakterisztikával rendelkező numerikus megoldásokat kaptam. Habár az összes feladatot teljesíteni nem volt időm, a megoldott feladatok során minden esetben az elvárt eredményeket kaptam. Az eddigiekkel ellentétben itt nem volt olyan mozzanat, amiről animációt lett volna érdemes készíteni, azonban ezt a sejtautomata esetében majd bepótolom.

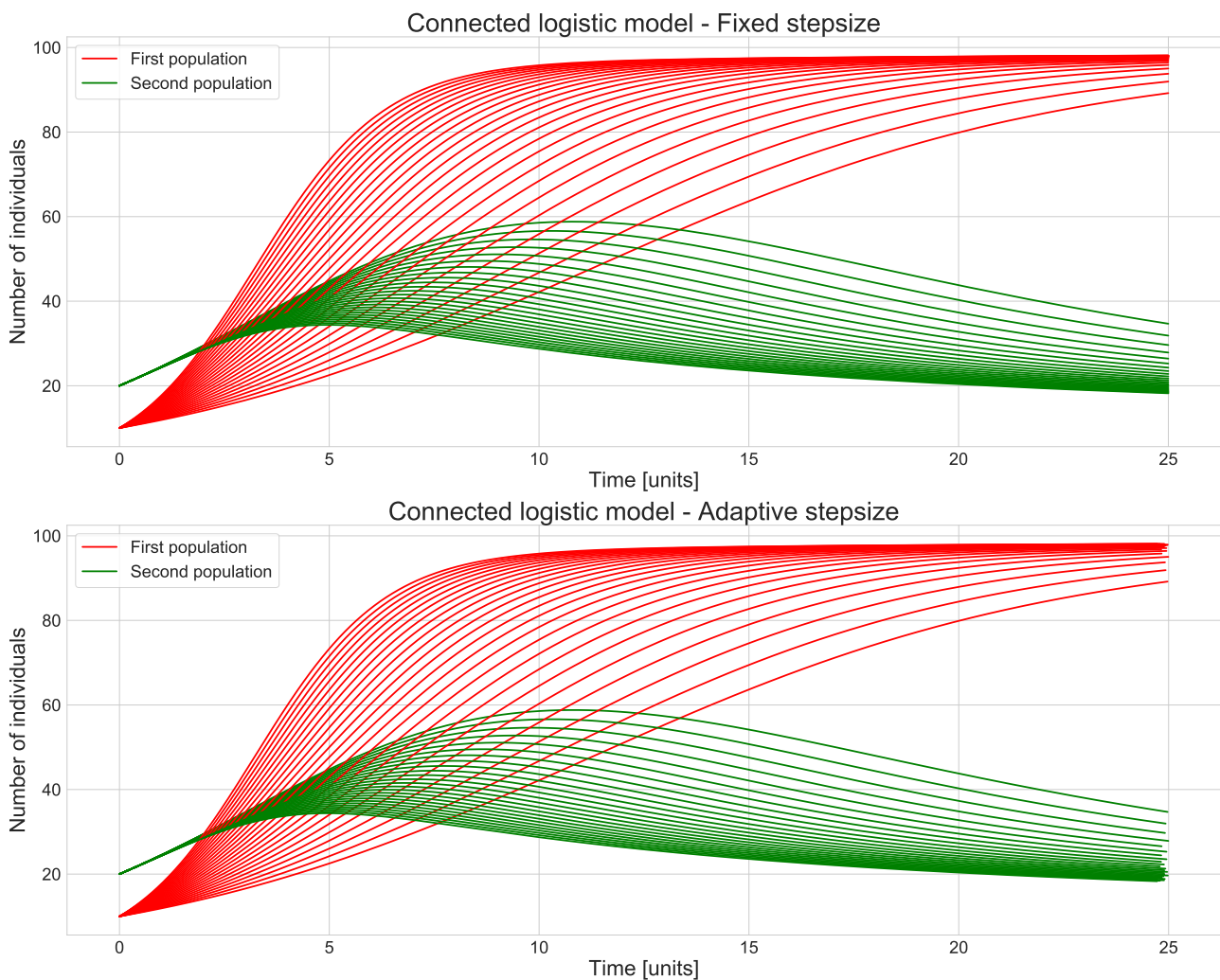
-
- [1] József Stéger, István Csabai. *Számítógépes szimulációk – Populációdinamika*. [Online; opened at April 25, 2019]. 2019. URL: <https://stegerjozsef.web.elte.hu/teaching/szamszim/popdin.pdf>.
 - [2] Pál Balázs. *ELTE Computer Simulations 2019 — GitHub*. [Online; opened at March 23, 2019]. 2019. URL: https://github.com/masterdesky/ELTE_Comp_Simulations_2019.

APPENDIX A - ÁBRÁK

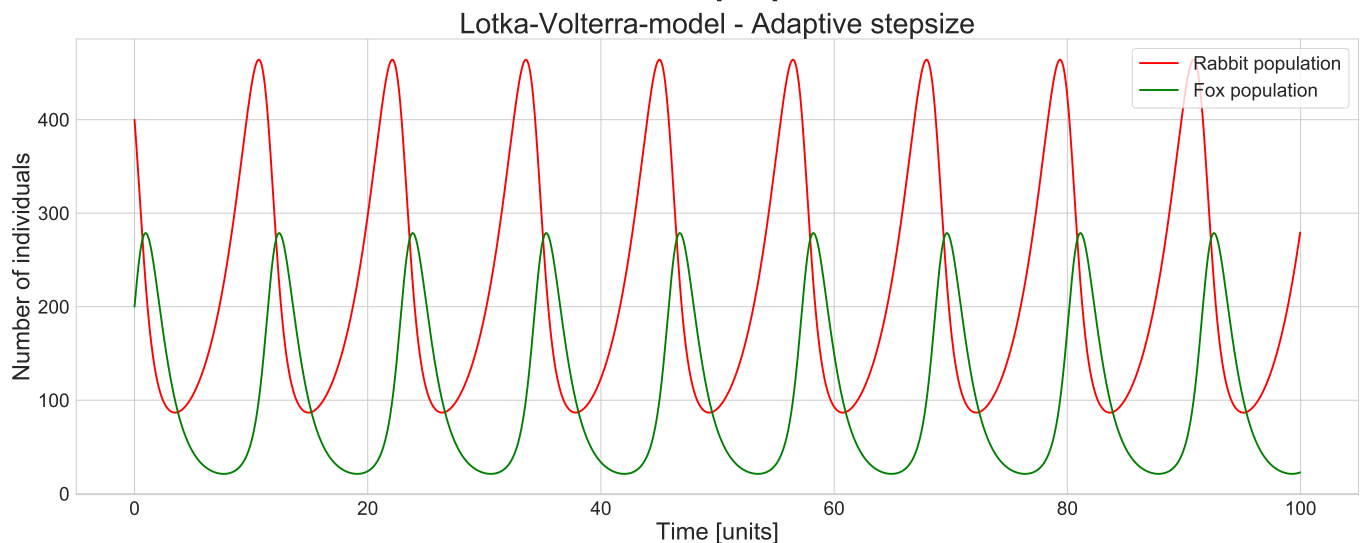
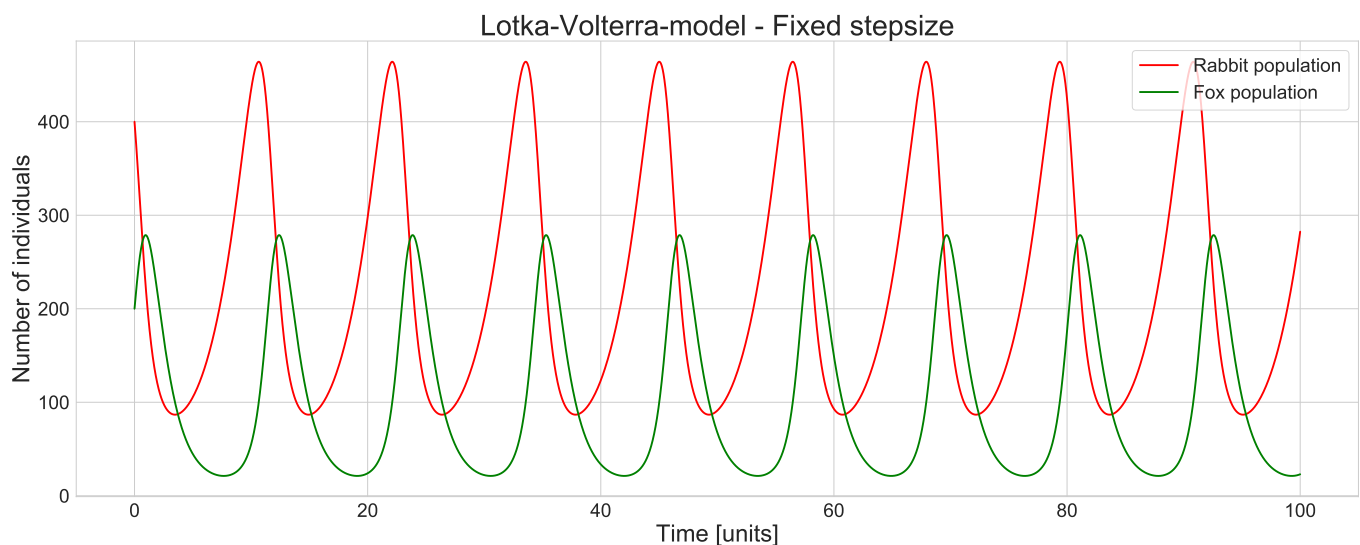
A.1. AZ VIZSGÁLT MODELLEK IDŐFEJLŐDÉSE



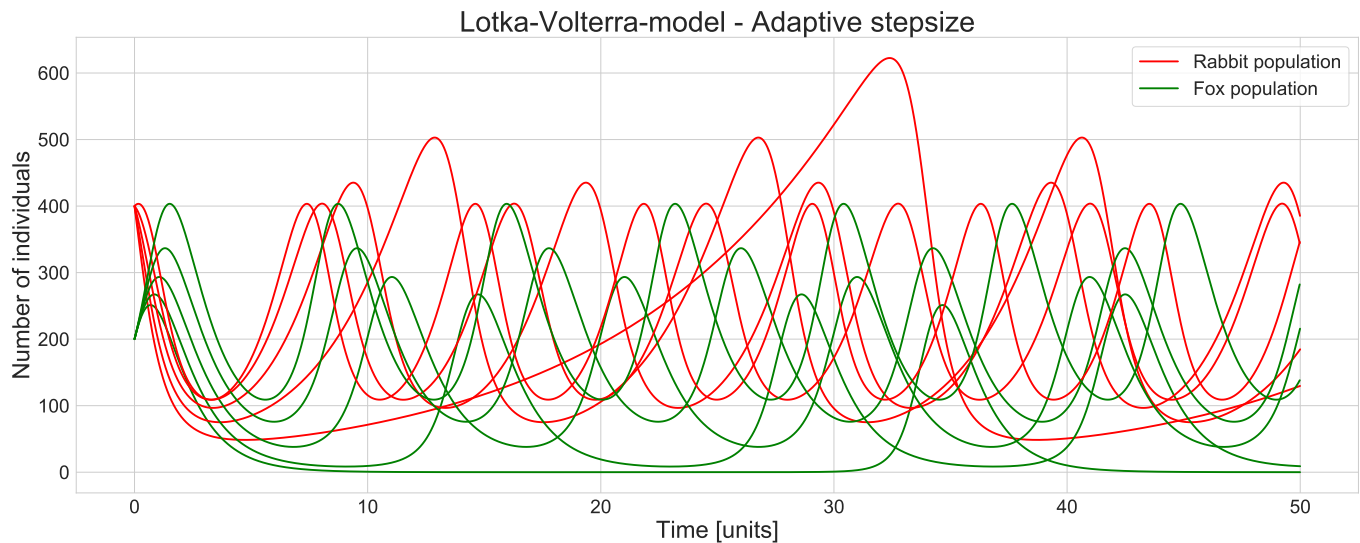
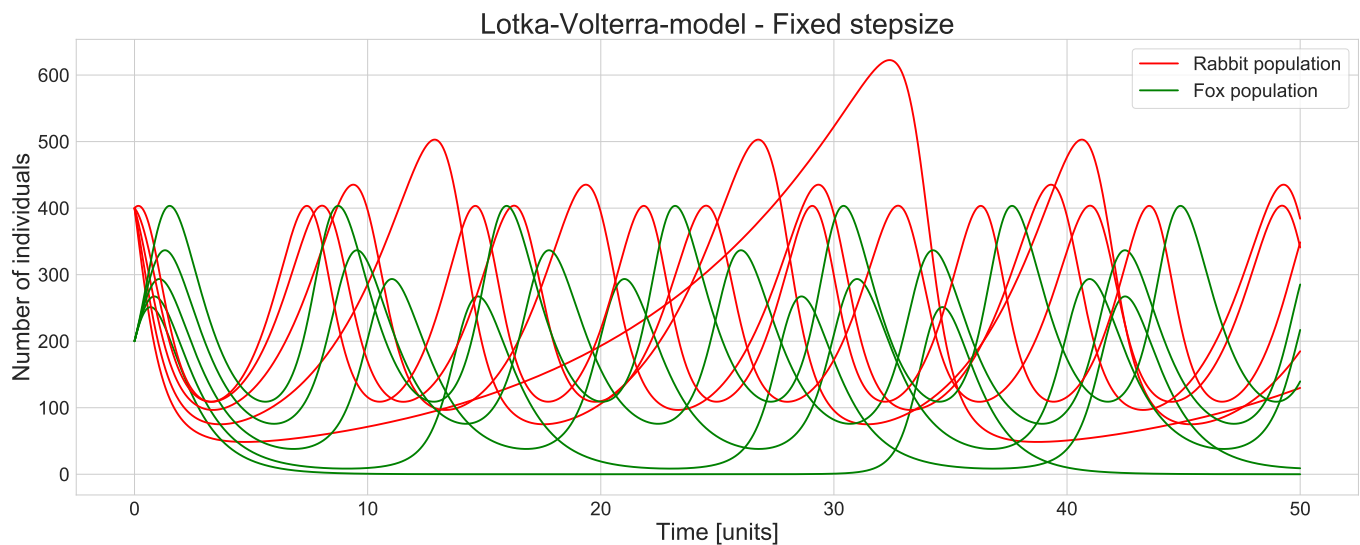
1. ábra. A logisztikus modell differenciálegyenletének megoldását adó egyik lehetséges görbesereg $w_{be} \in [0.5, 1]$ születési rátára, $w_{ki} = 0.3$ halálozási ráta, $n_0 = 10$ kezdeti egyedszám és $k = 100$ maximális egyedszám mellett.



2. ábra. A csatolt-logisztikus modell differenciálegyenletének megoldását adó egyik lehetséges görbesereg $w_{be1} \in [0.5, 1]$ -re, ahol w_{be1} a sorszám szerinti első faj születési rátája, $w_{be2} = 0.6$ születési, $w_{ki1} = w_{ki2} = 0.3$ halálozási ráták, $n_{01} = 10$, $n_{02} = 20$ kezdeti egyedszámok és $k_1 = k_2 = 100$ maximális egyedszám, valamint $\alpha = 0.1$ és $\beta = 0.9$ kölcsönhatási tényezők mellett.

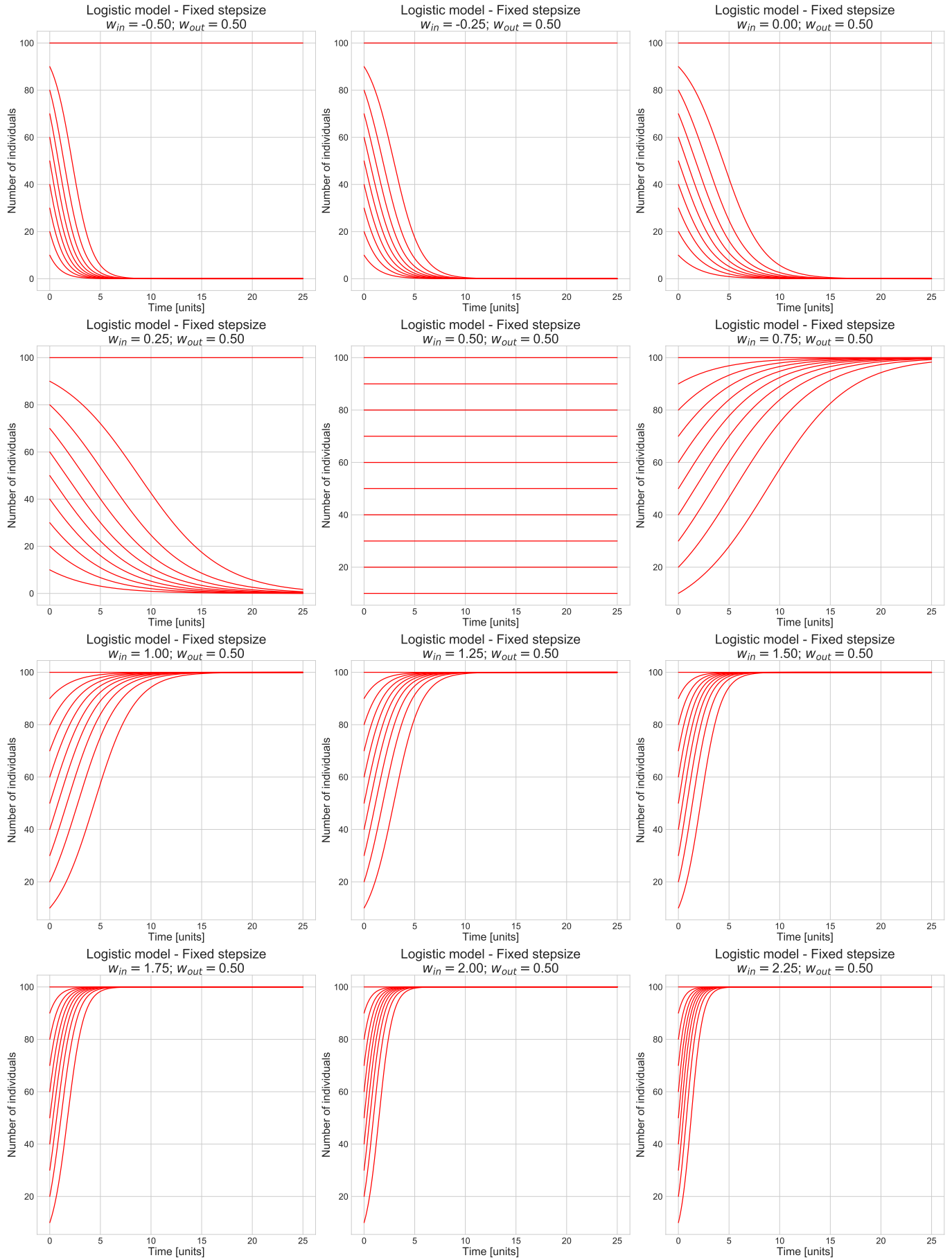


3. ábra. A Lotka–Volterra-modell differenciálegyenletének megoldását adó egyik lehetséges görbe, $n_{0,r} = 400$, $n_{0,f} = 200$ kezdeti egyedszámok, valamint $a = 0.4$, $b = c = 0.004$, $d = 0.9$ fejlődési ráták mellett, $k \rightarrow \infty$, $s = 0$ határesetben.

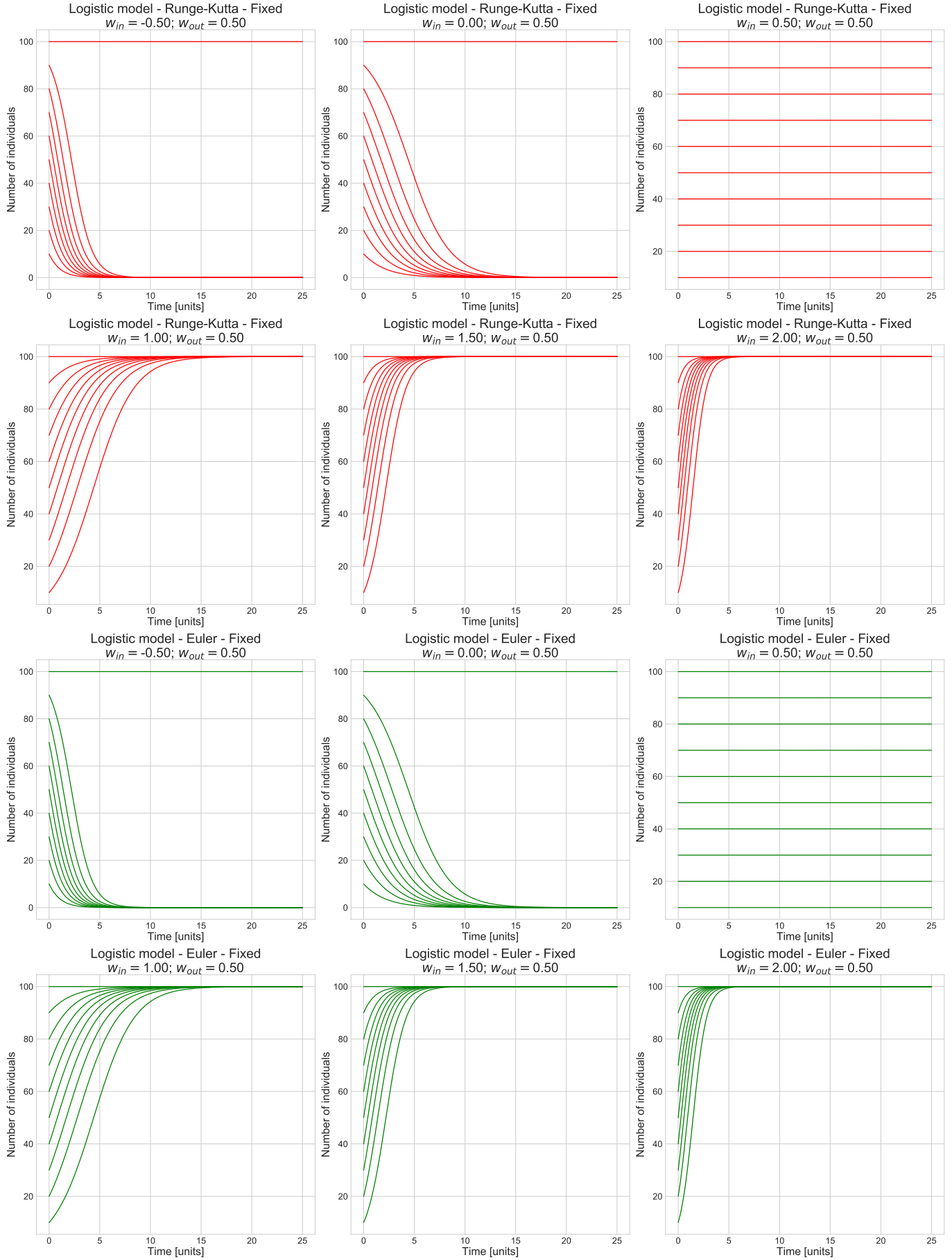


4. ábra. A Lotka–Volterra-modell differenciálegyenletének megoldását adó egyik lehetséges görbesereg, $a \in [0.1, 1.1]$ paraméter esetén, $n_{0,r} = 400$, $n_{0,f} = 200$ kezdeti egyedszámok, valamint $b = c = 0.004$, $d = 0.9$ fejlődési ráták mellett, $k \rightarrow \infty$, $s = 0$ határesetben.

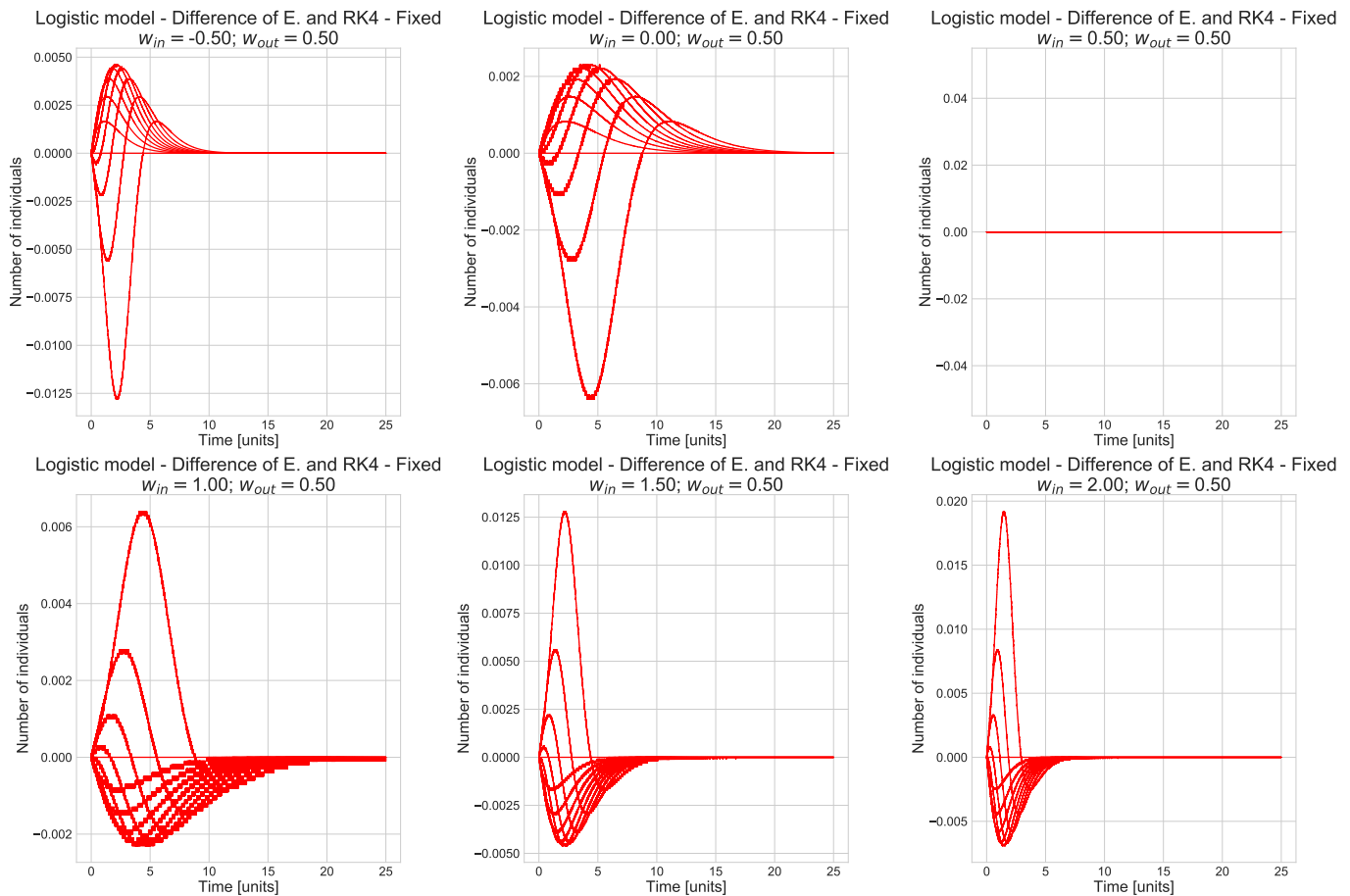
A.2. A LOGISZTIKUS-MODELL STABILITÁSÁNAK VIZSGÁLATA AZ $R = -W_{KI} + W_{BE}$ RÁTÁK ARÁNYÁNAK FÜGGVÉNYÉBEN



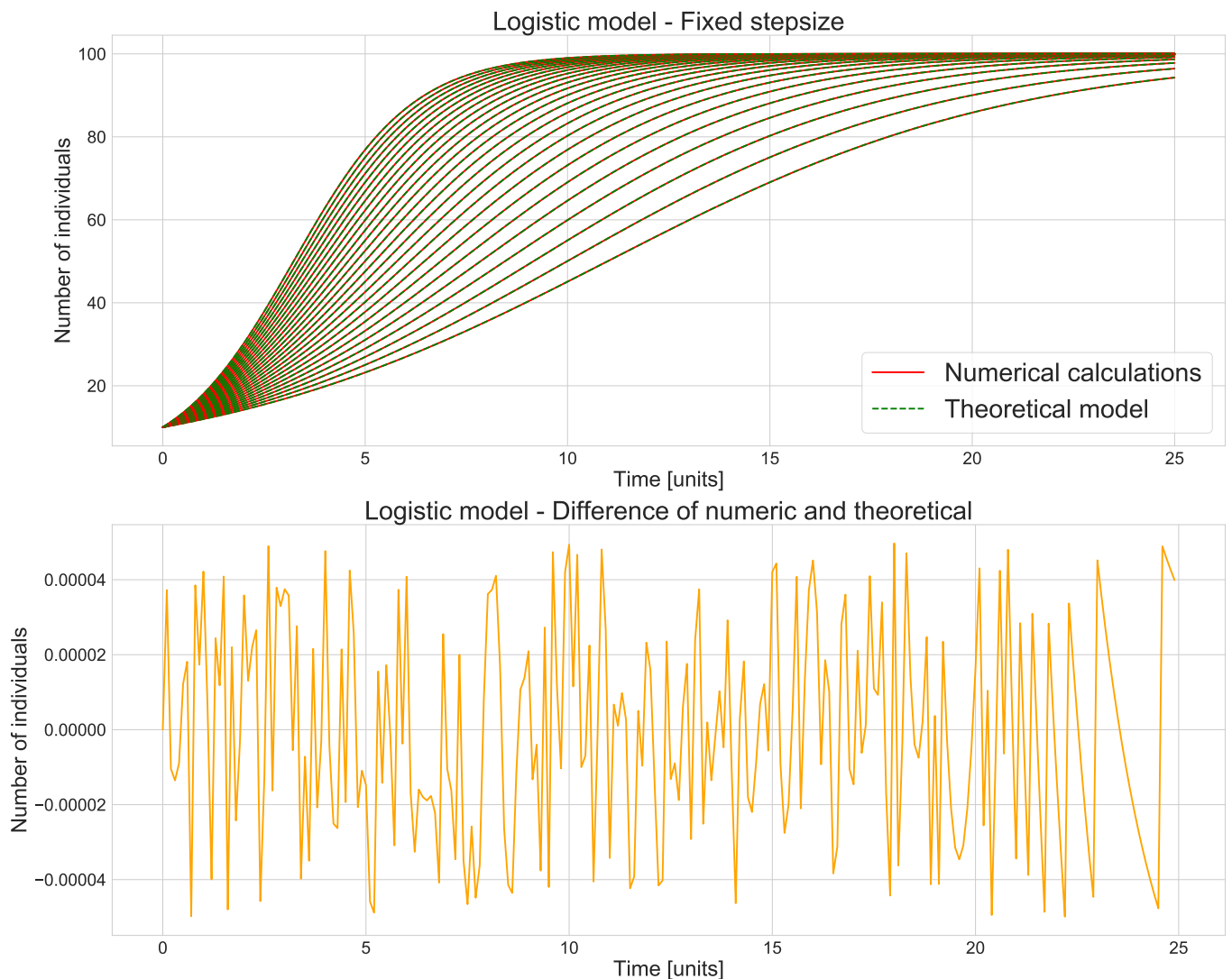
5. ábra. A logisztikus-modell stabilitásának vizsgálata, $w_{be} \in [-0.5, 2.25]$ születési, $w_{ki} = 0.5$ halálozás ráták mellett, minden egyes r érték esetén $n_0 \in [10, 100]$ kezdeti egyedszám és $k = 100$ maximális egyedszám esetében.



6. ábra. A logisztikus-modell stabilitásának vizsgálata és az Euler-, valamint negyedrendű Runge–Kutta-módszerek összehasonlítása, $w_{be} \in [-0.5, 2]$ születési, $w_{ki} = 0.5$ halálozás ráták mellett, minden egyes r érték esetén $n_0 \in [10, 100]$ kezdeti egyedszám és $k = 100$ maximális egyedszám esetében. A felső 6 grafikon a negyedrendű Runge–Kutta módszerrel integrált görbeseregeket, míg az alsó 6 az Euler-módszer segítségével integráltakat mutatja.

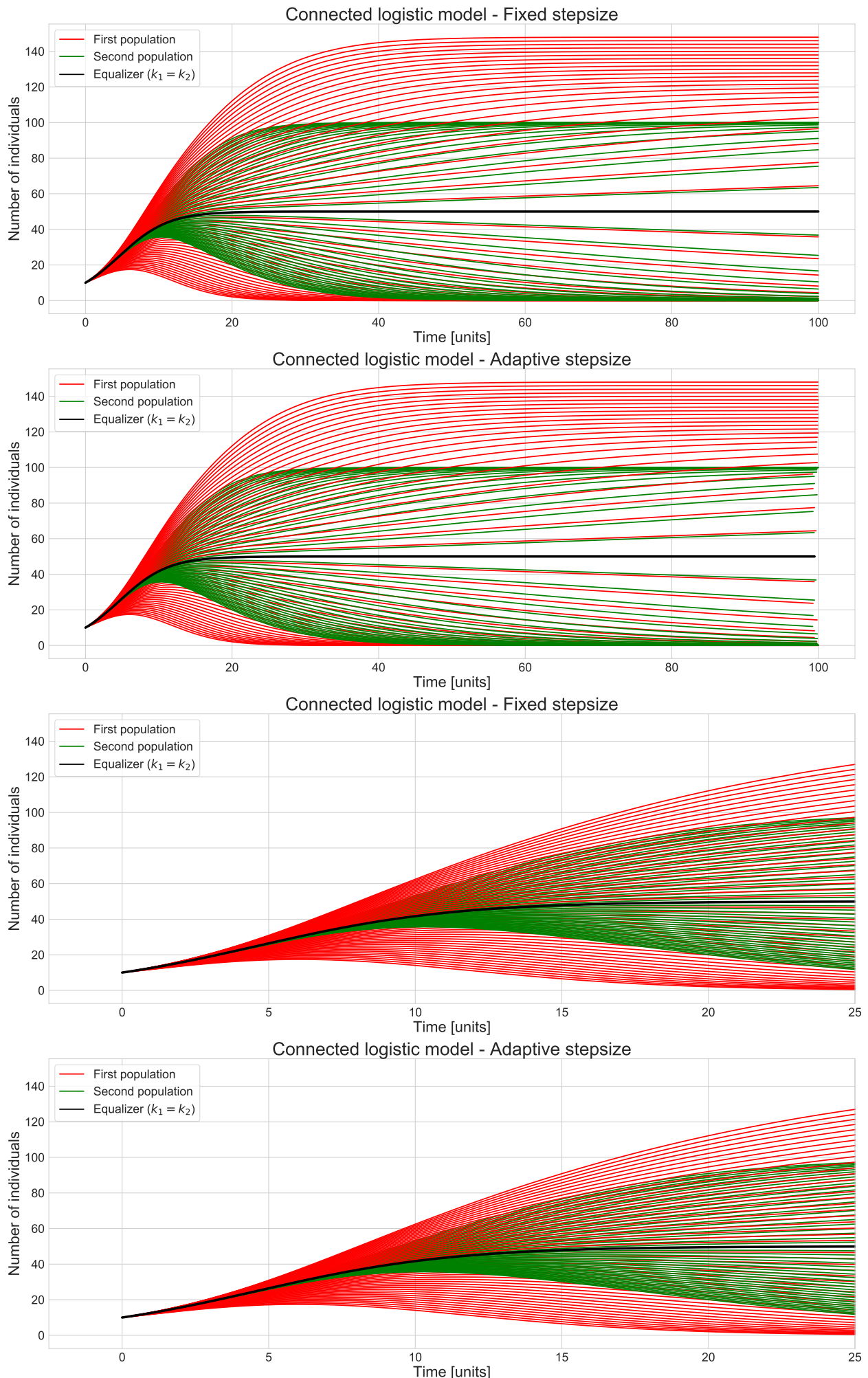


7. ábra. A logisztikus-modell stabilitásának vizsgálata és az Euler-, valamint negyedrendű Runge–Kutta-módszerek előző ábrán szereplő értékeinek különbsége, $w_{be} \in [-0.5, 2]$ születési, $w_{ki} = 0.5$ halálozás ráták mellett, minden egyes r érték esetén $n_0 \in [10, 100]$ kezdeti egyedszám és $k = 100$ maximális egyedszám esetében.



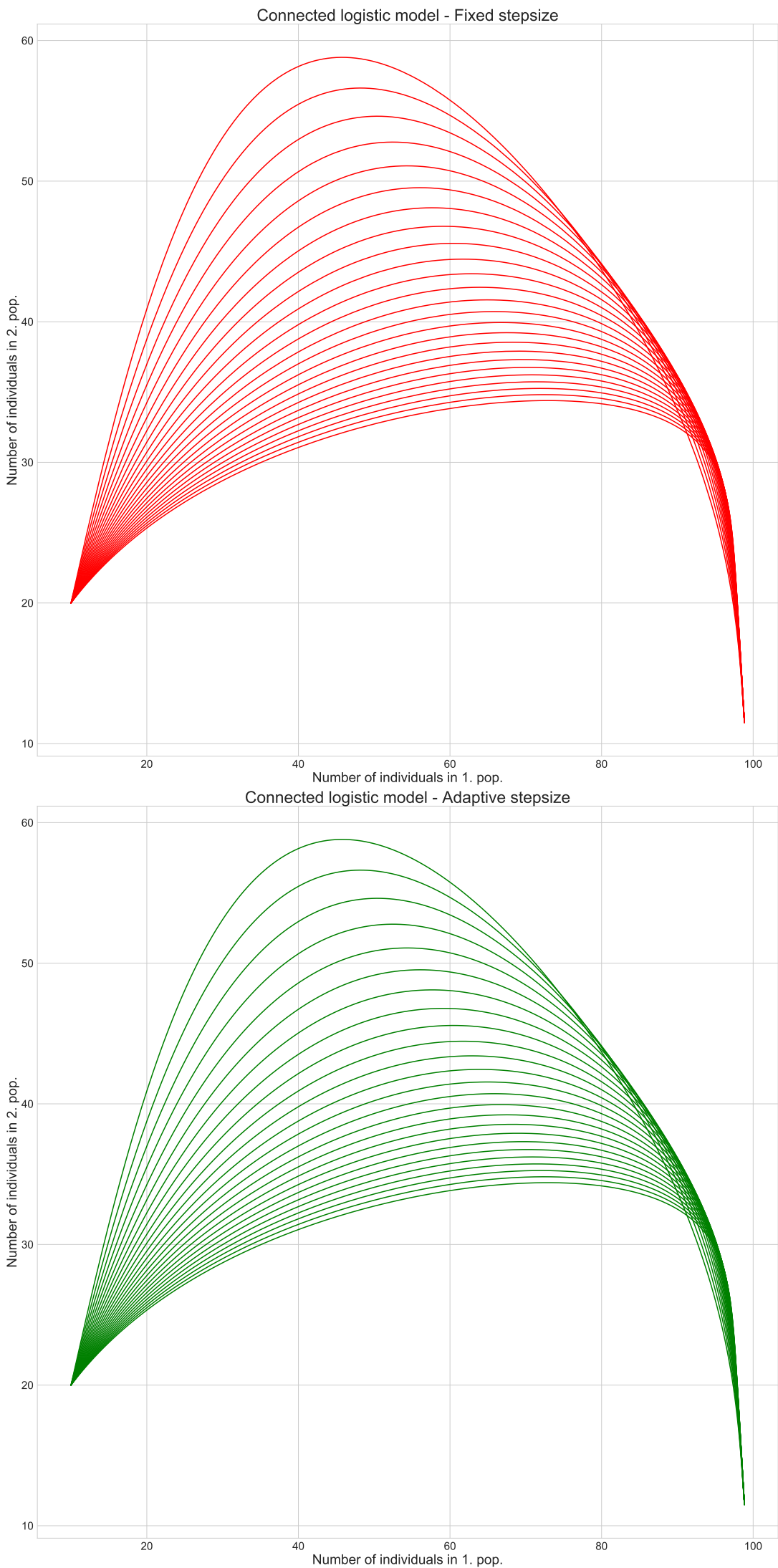
8. ábra. A logisztikus-modell numerikus és elméleti eredményeinek összehasonlítása, $w_{be} \in [0.5, 1]$ születési, $w_{ki} = 0.3$ halálozás ráták mellett, minden egyes r érték esetén $n_0 = 10$ kezdeti egyedszám és $k = 100$ maximális egyedszám esetében. Az alsó kép szemmel láthatóan magyarázatot igényel. Ezen a felette látható grafikonon található görbék közül legutolsóhoz ($w_{ki} = 1$) tartozó numerikus és elméleti eredmény különbségét ábrázoltam. A kép minden egyes görbére hasonló, zajra emlékeztető viselkedést mutatott. A jobb átláthatóság kedvéért az egyetlen ábrázolt görbének csak minden 100. pontját jelenítettem meg a grafikonon. Erről az olvasható le, hogy az elméletitől való eltérés szinte minden lépésben, két szélső pont között ugrial (jelen képen ezek kb. a ± 0.00004 értékek). A kapott eredmény mindenképp elgondolkodtató és egyértelműen valamilyen szisztematikus numerikus hibát jelent. Mivel a numerikus eredményt a Runge–Kutta–Cash–Karp-módszer felhasználásával készítettem, így a hiba mindenképp abban keresendő.

A.3. A CSATOLT-LOGISZTIKUS-MODELL STABILITÁSÁNAK VIZSGÁLATA A K MAXIMÁLIS EGYEDSZÁM FÜGGVÉNYÉBEN

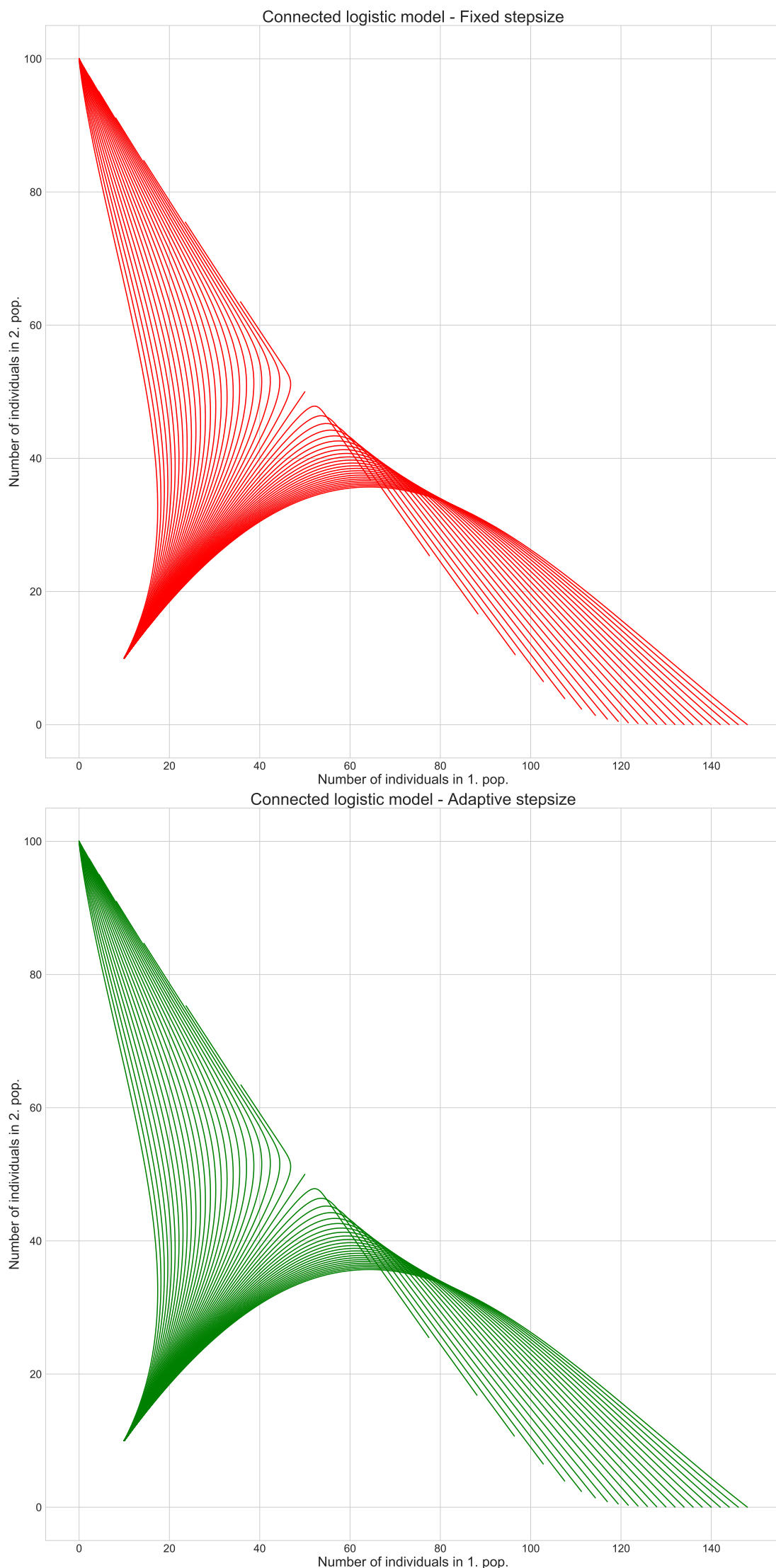


9. ábra. A csatolt-logisztikus modell stabilitásának vizsgálata, $k_1 \in [50, 150]$ -re, $k_2 = 100$ maximális egyedszám, $w_{be_1} = w_{be_2} = 0.6$ születési és $w_{ki_1} = w_{ki_2} = 0.3$ halálozási ráták, $n_{0_1} = n_{0_2} = 10$ kezdeti egyedszám, valamint $\alpha = \beta = 1$ kölcsönhatási tényezők mellett. A vastag fekete sáv a $k_1 = k_2$ esetnek megfelelő görbét jelöli, mely esetében a rendszer instabil egyensúlyi helyzetben tartózkodik. A feladatok alapján elemeznünk kellett, hogy igaz-e az állítás, miszerint a nagyobb k paraméterű faj győzedelmeskedik a csatolt-logisztikus egyenlet alapján? Az ábra pontosan ezt a viselkedést próbálja szemléltetni. A k_1 értéket folyamatosan változtatva k_2 -t szimmetrikusan körbepáztáztam, majd a hozzájuk tartozó görbéket ábrázoltam. Ez alapján kaptam meg a fenti grafikonokat. Ezekről az olvasható le, hogy a k_1 növelésével a piros (1-es állatfaj) egyre meredekebben növekedik kezdetben, míg a 2-es állatfaj egyre inkább csökken. Mikor k_1 érték megegyezik k_2 -vel, a rendszer egyensúlyba kerül, azonban bármerre is mozdul ki onnan, minden esetben divergál. Amíg k_2 nagyobb értékkel rendelkezik, addig a 2-es populáció száma konvergál a maximumértékéhez, míg az 1-es 0-hoz tart, azonban amint a k_1 kerekedik felül, a helyzet megfordul. Az alsó két grafikonon a fentiek első negyede van kiemelve.

A.4. A CSATOLT-LOGISZTIKUS-MODELL FÁZISDIAGRAMJAI

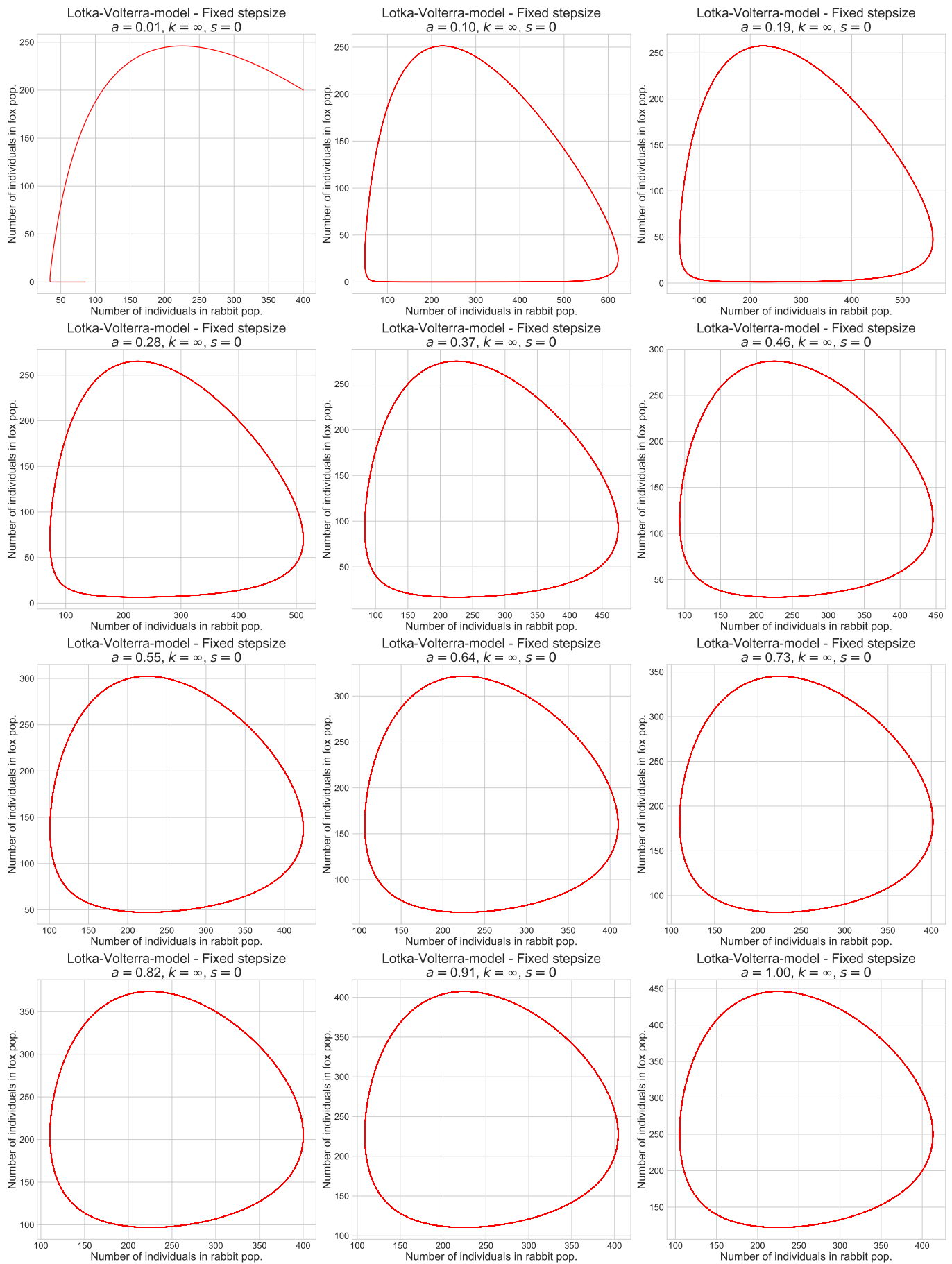


10. ábra. A csatolt-logisztikus modell populációinak változása, $n_1 - n_2$ diagramon ábrázolva, $w_{be_1} \in [0.5, 1]$ -re, ahol w_{be_1} a sorszám szerinti első faj születési rátája, $w_{be_2} = 0.6$ születési, $w_{ki_1} = w_{ki_2} = 0.3$ halálozási ráták, $n_{0_1} = 10$, $n_{0_2} = 20$ kezdeti egyedszámok és $k_1 = k_2 = 100$ maximális egyedszám, valamint $\alpha = 0.1$ és $\beta = 0.9$ kölcsönhatási tényezők mellett, $\text{sim_time} = 100$ időegység hosszú szimuláció esetén.

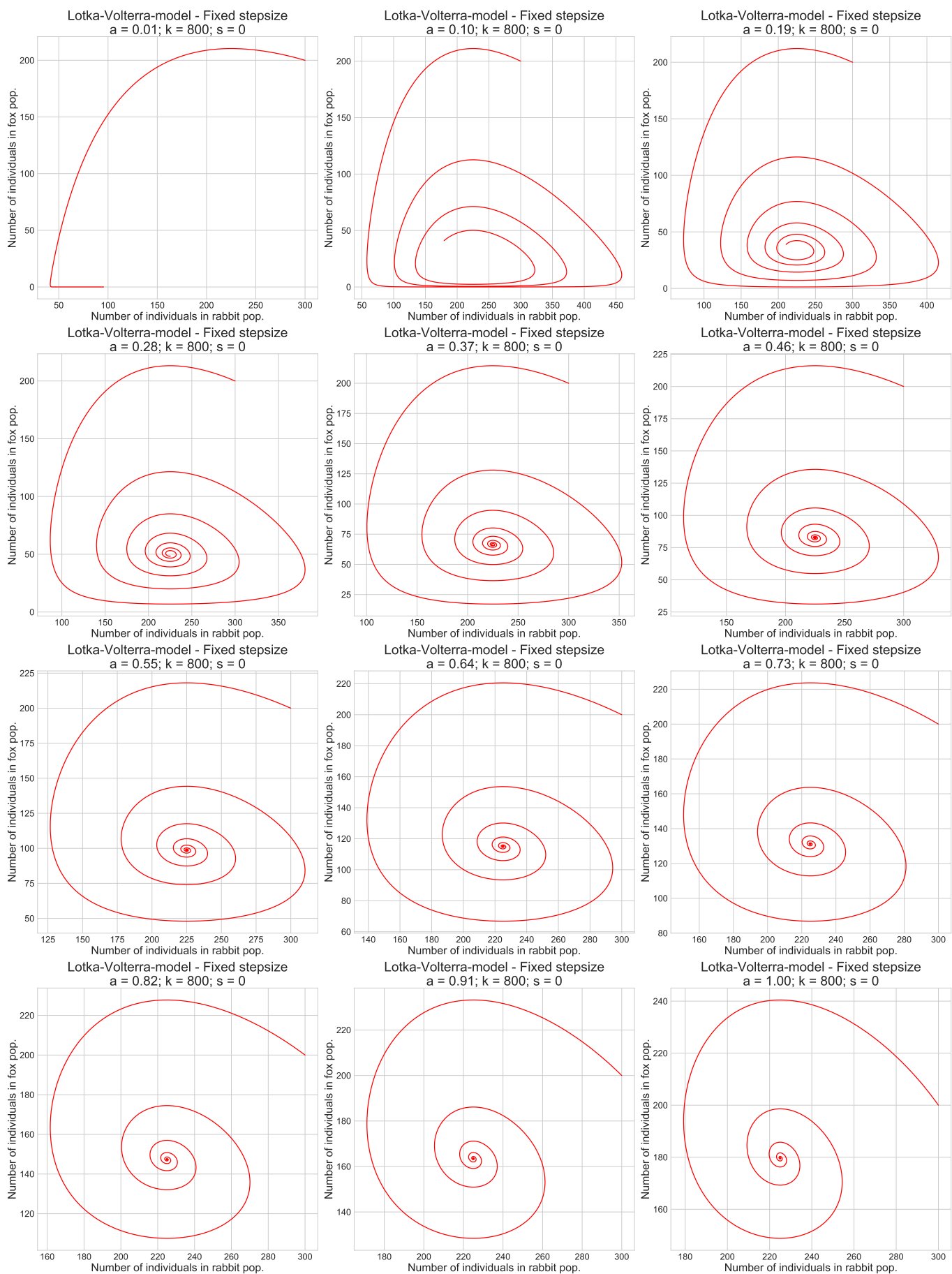


11. ábra. A csatolt-logisztikus modell populációinak változása, $n_1 - n_2$ diagramon ábrázolva, $k_1 \in [50, 150]$ -re, $k_2 = 100$ maximális egyedszám, $w_{be_1} = w_{be_2} = 0.3$ születési és $w_{ki_1} = w_{ki_2} = 0.3$ halálozási ráták, $n_{0_1} = 10$, $n_{0_2} = 20$ kezdeti egyedszámok, valamint $\alpha = \beta = 1$ kölcsönhatási tényezők mellett, $\text{sim_time} = 100$ időegység hosszú szimuláció esetén.

A.5. A LOTKA–VOLTERRA-MODELL FÁZISDIAGRAMJAI



12. ábra. A Lotka–Volterra-modell populációinak változása, $n_r - n_f$ diagramon ábrázolva, $a \in [0.1, 1.1]$ paraméter esetén, $n_{0_r} = 400$, $n_{0_f} = 200$ kezdeti egyedszámok, valamint $b = c = 0.004$, $d = 0.9$ fejlődési ráták mellett, $k \rightarrow \infty$, $s = 0$ határesetben, $\text{sim_time} = 100$ időegység hosszú szimuláció esetén.



13. ábra. A Lotka–Volterra-modell populációinak változása, $n_r - n_f$ diagramon ábrázolva, $a \in [0.1, 1.1]$ paraméter esetén, $n_{0_r} = 300$, $n_{0_f} = 200$ kezdeti egyedszámok, valamint $b = c = 0.004$, $d = 0.9$ fejlődési ráták mellett, $k = 800$, $s = 0$ esetben, $\text{sim_time} = 100$ időegység hosszú szimuláció esetén.