

Diszkrét elemes módszerek implementációja grafikus processzorokra numerikus talajszimulációk gyorsításához

Implementation of discrete element methods on GPUs to speed up numerical soil simulations

NAGY Dániel¹, PÁSTHY László², TAMÁS Kornél²

¹Budapesti Műszaki és Gazdaságtudományi Egyetem, Gépészmérnöki Kar,
Hidrodinamikai Rendszerek Tanszék
H-1111, Budapest, Műegyetem rkp. 3. Tel.: +361-463-1646.
E-mail: dnagy@hds.bme.hu, Web: <https://www.hds.bme.hu>

²Budapesti Műszaki és Gazdaságtudományi Egyetem, Gépészmérnöki Kar,
Gép- és Terméktervezés Tanszék
H-1111, Budapest, Műegyetem rkp. 3. Tel.: +361-463-3511.
E-mail: pasthy.laszlo@gt3.bme.hu, tamas.kornel@gt3.bme.hu Web: <http://www.gt3.bme.hu>

Abstract

This work presents the in-house-developed GPU-based discrete element method (DEM) solver, which is more than an order of magnitude faster than other available DEM software due to GPU acceleration. The goal is to use the program to create simulations of the soil-sweep interactions and to reproduce measurement results in wet sand soil with automatic parameter calibration using an evolutionary algorithm.

Keywords: discrete element methods, GPU, high-performance computing, soil simulation, optimization

Kivonat

Ez a munka a saját fejlesztésű GPU alapú diszkrét elemes megoldót (DEM) mutatja be, amely a GPU-s gyorsításnak köszönhetően több mint egy nagyságrenddel gyorsabb mint más elérhető DEM szoftverek. A cél a program felhasználása kultivátoros talajszimulációk készítéséhez, és homok talajban végzett mérési eredmények reprodukálása automatikus paraméterkalibrációval egy evolúciós algoritmus segítségével.

Kulcsszavak: diszkrét elemes módszerek, GPU, nagy teljesítményű számítástechnika, talajszimuláció, optimalizáció

1. BEVEZETÉS

A talaj megfelelő gondozása és művelése kiemelt fontosságú az emberiség számára, mivel a talaj minőségének megőrzése és a fenntartható gazdálkodás lehetővé teszi az élelmiszertermelés növelését és a természeti erőforrások megóvását. Bár a talajművelés több ezer éves tudomány, azonban mind a mai napig tart a technológia fejlődése és jelennek meg újabb talajművelő eszközök. Napjainkban a környezetbarát talajművelés egyre inkább előtérbe kerül talajművelő eszközök tervezése esetén, ennek lényege az energiatakarékosság, a talajerózió szabályozása és a károsanyag-kibocsátás csökkentése [1]. A megfelelő talajművelő eszköz megtervezése, sőt még kiválasztása is nehéz feladat, hiszen ez a talajtípustól függően eltérhet. A talaj-szerszám kölcsönhatások matematikai modellezése lehetővé teszi a korábban ismert problémák elméleti megoldását, például lehetőség nyílik egy kultivátor numerikus optimalizációjára, az energiahatékonyság növelése érdekében. Az utóbbi másfél évtizedben elterjedt a diszkrét elemes módszer (DEM) alkalmazása talajszimulációkra. A DEM előnye, hogy segítségével figyelembe vehető a talaj inhomogenitása és lehetőség nyílik a talajszemcsék áramlásának a meghatározására is [2]. Diszkrét elemes módszerrel számos mezőgazdaságban jelentkező probléma szimulálható és megoldható, többek között lehetséges a vonóerő meghatározásán túl egy kultivátoros talajművelés során kialakuló repedések vizsgálata is [3]. A fő cél egy saját fejlesztésű moduláris GPU alapú DEM megoldó készítése CUDA C++-ban, amely megbízhatóan használható sok szemcsét tartalmazó talajszimulációk készítésére és a GPU-s gyorsításnak

köszönhetően legalább egy nagyságrenddel gyorsabb, mint a meglévő szoftverek. Feltétel, hogy a programban tetszőleges számú és méretű szemcse, anyag és peremfeltétel megadható legyen, illetve legalább egy mozgatható geometria is hozzáadható legyen, amely a mezőgazdasági alkalmazás miatt fontos.

2. A PROGRAM FELÉPÍTÉSE

Az elmúlt évek fejlesztései lehetővé teszik, hogy a GPU-kat rendkívül komplex problémák megoldására is használjuk, így például diszkrétélemes szimulációkra. A GPU-k előnye abban rejlik, hogy több ezer számítási egységet tartalmaznak, amelyek képesek ugyanazt a műveletet, egyszerre, párhuzamosan elvégezni, más-más adatokon.

A teljes DEM program CPU, GPU és fordítási időben kiszámolt részt is tartalmaz. A programcsomag fordításának idejében már ismertnek kell lennie a szemcsék és anyagok számának, a választott kapcsolati modellnek, kapcsolatkeresési algoritmusnak és időléptetési módszernek. Ezek lehetővé teszik a programkód fordító általi optimalizálását, növelve annak teljesítményét. A CPU fő feladata a kezdeti feltételek beolvasása, anyagparaméterek beállítása, a memória kezelése és a kimeneti fájlok elkészítése. A GPU feladata a diszkrétélemes algoritmus megvalósítása, amely a következő lépésekből áll:

1. Szemcse-szemcse érintkezések detektálása: Feltételezve, hogy a szemcsék gömb alakúak, a szemcsék közötti kapcsolatok könnyen detektálhatók. Két szemcse érintkezik, ha középpontjaik távolsága kisebb, mint sugaraik összege. Ez minden szemcsepárra kiszámolható. Az algoritmus lényegesen felgyorsítható, amennyiben a számítási tartomány alá egy hálót helyezünk és csak az azonos, vagy szomszédos cellában lévő szemcsék között kerül számításra a távolság [4].
2. Szemcse-geometria érintkezések detektálása: A geometria STL formátumban kerül megadásra, az STL fájlok háromszög által határolt geometriákat írnak le. A szemcse-geometria érintkezések megtalálásához tehát egy gömb és egy háromszög közötti érintkezés detektálása szükséges. Ez hatékonyan megvalósítható, mivel minden háromszöghöz egy saját koordináta-rendszer rendelhető, és a szemcse középpontját a háromszöghöz rendelt koordináta-rendszerben felírva az érintkezés megállapítható.
3. Kapcsolati modell: A szemcsékre ható erők meghatározása egy adott érintkezési modell szerint. Jelenleg a Hertz-Mindlin modell került implementálásra [5], amely figyelembe veszi a szemcsék összenyomódásából adódó erőt és kiegészíti ezt a tangenciális merevséggel, ez ma az egyik legelterjedtebb kapcsolati modell a DEM-ben. A Hertz-Mindlin modell alapja a szemcseütközés során az átfedés hatására fellépő erők meghatározása.
4. Időlépés: A szemcsére ható erőkből gyorsulások számítása és a szimuláció időbeni előreléptetése a megfelelő numerikus módszerekkel. Ehhez implementálásra került az egy lépéses explicit Euler módszer, amelyet a szakirodalomban is jellemzően alkalmaznak.

A kezdeti szemcse eloszlás beolvasása és a szemcsék bizonyos időközönkénti elmentése elengedhetetlen. A program alapvetően *vtu* kiterjesztésű fájlok olvasására és írására képes. A *vtu* fájlokat a nyílt-forráskódú VTK Visualization Toolkit definiálja. A fájl típus előnye, hogy közvetlenül beolvasható Paraview-ba és hozzáadható minden szemcséhez bármilyen további tetszőleges tulajdonság, például a szemcsék sebessége, a szemcsékre ható erők vagy akár a szemcsék anyaga.

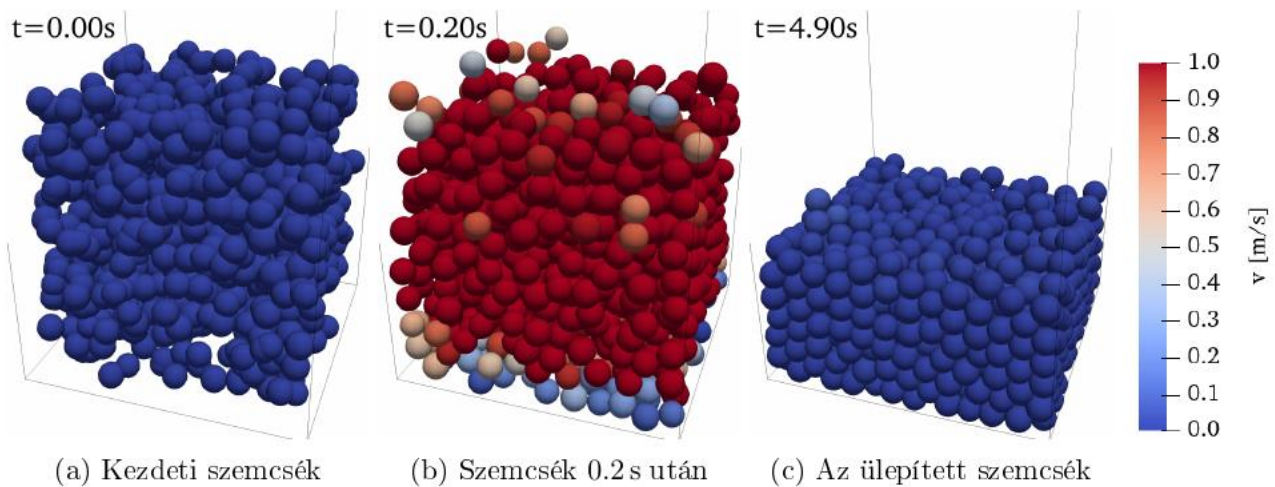
A saját fejlesztésű program összehasonlításra került egy 8519 szemcséből álló ülepítési problémán, az Altair EDEM® szoftverrel. Mindkét esetben ugyanazon kezdeti szemcseeloszlás került beállításra, ugyanolyan paraméterekkel. A szimulációk során a szemcsehalmoz számolt kinetikus és potenciális energiája megegyezett, illetve a szimulációk kvalitatív módon is megegyeztek, ami validálja a saját fejlesztésű program eredményeinek a helyességét. Emellett, a saját fejlesztésű program 12-szer gyorsabbnak bizonyult az Altair EDEM®-nél.

3. A PROGRAM ALKALMAZÁSA

3.1. Ülepítés

Az elkészült program felhasználásra került először gravitációs ülepítésre további validáció céljából, majd kultivátoros talajszimulációkra. Az ülepítés lényege egy megfelelő kezdeti szemcseeloszlás létrehozása további szimulációkhoz. Gravitációs ülepítés esetén a szemcsék kezdetben véletlenszerű helyekre kerülnek egy adott tartományban, majd a gravitáció hatására leesnek és egymásra rétegződnek. Az ülepítés során elvárás, hogy minden szemcse sebessége nullához tartson. Az ülepítésre egy példa az 1. ábrán látható, kezdetben a szemcsepozíciók véletlenszerűen kerültek generálásra egyenletes eloszlás szerint a megadott tartományban. Az ábrán megfigyelhető, hogy kezdetben a szemcse sebességek növekednek, hiszen a szemcsék

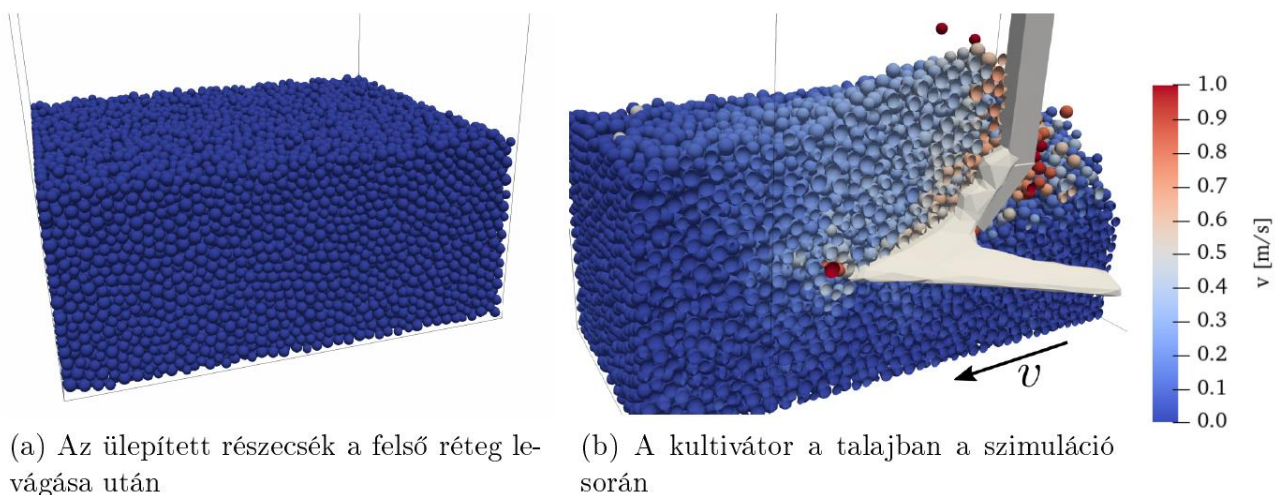
esés közben folyamatosan gyorsulnak, a szimuláció végén 4.9 másodpercnél az 1c ábra szerint a szemcsék sebessége már közel nulla. Valójában a sebességek nem csökkennek sosem pontosan nullára a véges időlépésből és a kerekítési hibákból adódóan.



1. ábra: 1024 szemcse ülepitésének a szimulációja a saját fejlesztésű programban. Az ábrák színezése a sebességvektor nagysága alapján történt a jobb oldalt látható skála szerint.

3.2 Kultivátoros talajszimulációk

A kultivátoros talajművelés célja a talaj fellazítása, ami növeli a vízfelvevő képességet és lehetővé teszi a gyökerek könnyebb behatolását a talajba. A következő példa célja bemutatni, hogy a saját fejlesztésű DEM program alkalmazható gyakorlati problémák vizsgálatára is, mint például a kultivátorra ható erők meghatározására a talajművelés során. A cél pedig mérési eredmények reprodukálása a DEM paraméterek kalibrációja által. A laboratóriumi talajvályús mérés során egy homok talajban került végighúzásra egy kultivátor $v = 0.7$ m/s sebességgel és a kísérlet során $F = 309.5$ N átlagos vonóerőt mértek [7]. A homok talaj tömeg alapú nedvességtartalma 3.96% volt a mérésben. A szimulációk során a mérésekben használt kultivátor modell 3D szkennelt változata került alkalmazásra.



2. ábra: Létrehozott szemcsehalmaz az ülepités után és a talajművelés során. A tartomány mérete 0.7×0.6 méter. A színezés a sebességvektor nagyságával arányos a jobb oldali skála szerint.

A mérés reprodukálása evolúciós algoritmussal történt, ehhez a szimulációk automatizálásra kerültek. Az automatizált szimuláció a következő lépésekből áll:

1. Kezdetben 38912 darab, $R = 8.5 \text{ mm} \pm 1.5 \text{ mm}$ sugarú szemcse (egyenletes eloszlással) kerül ülepítésre egy 0.7×0.6 méteres tartományban, amely szemcsék a paraméterek függvényében nagyjából 30–40 cm vastagon rétegződnek.
2. A szemcsékből eltávolításra kerül mindegyik, amely középpontja 30 cm felett van, így minden esetben egy hasonló 30 cm vastag talajréteg kerül létrehozásra, a paraméterektől függetlenül. Az ülepített szemcsék a 2a ábrán láthatók.
3. Az utolsó lépés a kultivátor végighúzása $v = 0.7 \text{ m/s}$ -el és az erők összegzése a szimuláció során. A vonóerő a $0.4 \text{ s} < t < 0.8 \text{ s}$ tartományban kerül kiátlagolásra, így a tartomány szélei nem befolyásolják a szimuláció eredményeit. A kultivátor végighúzásáról egy pillanatkép látható a 2b ábrán. A beállított munkamélység a méréseknek megfelelően 15 cm.

A felsorolt lépések futási ideje egy paraméterkombinációra mindössze 10-30 másodperc között adódik, így nagy mennyiségű szimuláció futtatható viszonylag rövid idő alatt. Paraméterkalibrációhoz a differenciál-evolúció került alkalmazásra, amely folytonos problémákra alkalmazható adott megszorítások mellett [6]. A differenciál evolúció lényege, hogy az új egyedek a meglévő egyedek lineáris kombinációjaként kerülnek generálásra, majd ezen új egyedek kerülnek keresztezésre a meglévőkkel. Az differenciál evolúcióval optimalizálandó mikromechanikai DEM paraméterek a szemcsesűrűség, a mikromechanikai Young-modulus, a mikromechanikai Poisson szám, az ütközési, csúszási súrlódási és a gördülési együttható. Minden egyed (\mathbf{x}) ezt a 6 kontrollváltozót tartalmazza, és összesen 60 egyed került létrehozásra kezdetben. A fitnessz függvény $f(\mathbf{x})$, azaz a minimalizálandó függvény a mérési értéktől való eltérés a vonóerőben, azaz

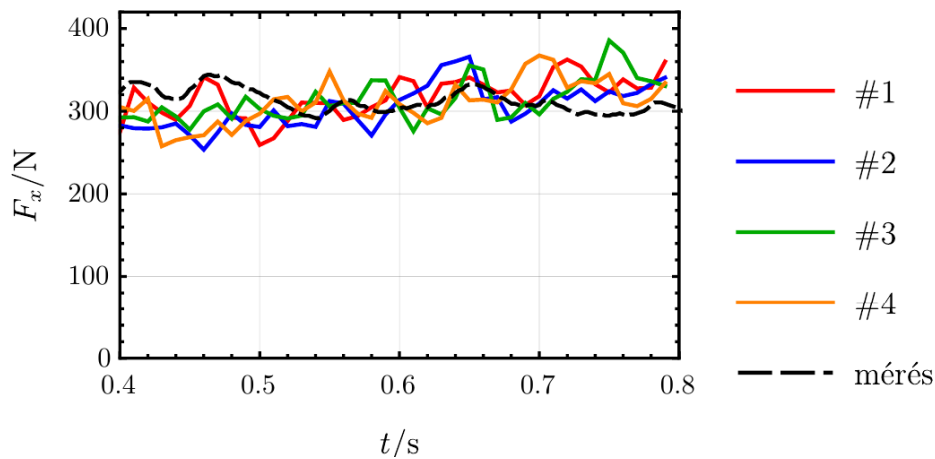
$$f(\mathbf{x}) = |F(\mathbf{x}) - 309.5 \text{ N}| \quad (1)$$

ahol $F(\mathbf{x})$ az átlagos vonóerő a szimulációban a megadott kontrollparaméterekkel. Az egyedek 44 generáció alatt konvergáltak be több különböző lokális minimumba. Ezen lokális minimumok a futtatás után K-Means klaszterezéssel kerültek megállapításra. A teljes futási idő 16 óra, ennyi idő alatt 2700-szor került végrehajtásra több mint 30000 szemcse ülepítése, majd a kultivátor végighúzása.

A megtalált lokális minimumokhoz tartozó mikromechanikai DEM paraméterek

1. táblázat

Kontrollváltozó	#1	#2	#3	#4
ρ szemcsesűrűség	2082 kg/m ³	2379 kg/m ³	1993 kg/m ³	2387 kg/m ³
E Young-modulus	$2.979 \cdot 10^6 \text{ Pa}$	$2.305 \cdot 10^6 \text{ Pa}$	$8.960 \cdot 10^6 \text{ Pa}$	$7.978 \cdot 10^6 \text{ Pa}$
ν Poisson-szám	0.259	0.475	0.227	0.486
e ütközési együttható	0.063	0.258	0.104	0.113
μ súrlódási együttható	0.684	0.637	0.684	0.495
μ_r gördülési együttható	0.229	0.277	0.282	0.264
egyedek száma a klaszterban	5	4	9	10
\bar{F}_x vonóerő	316.0 N	303.9 N	312.5 N	310.5 N
ΔF_x vonóerő szórása	25.6 N	26.2 N	25.5 N	25.8 N



3. ábra: A kultivátorra ható erők a kultivátoros talajlazítás során a különböző paraméterkombinációk mellett. A fekete szaggatott vonal a mérés eredményét ábrázolja. A paraméterek az 1. táblázatban láthatók.

A lokális minimumok paramétereit az 1. táblázat oszlopai tartalmazzák, minden esetben a szimulált vonóerő 300 és 320 N között adódik. Megfigyelhető, hogy meglehetősen különböző paraméterekkel is elérhető hasonló vonóerő, a megadott sebesség mellett. A 3. ábrán a szimulált vonóerő került ábrázolásra a négy különböző paraméterkombinációra és a mért vonóerő is feltüntetésre került szaggatott fekete vonallal. Minden paraméterkombinációra jó egyezés figyelhető meg a mérés és a szimuláció között az átlagerőre, azonban a szimulációkban nagyobb fluktuáció mutatkozik, mint a mérésben, ami a szimuláció diszkrét jellegével magyarázható, ugyanis a szimulációban a gömbök mérete meghaladta a valós talajszemcsék méretét. Az optimalizáció továbbfejleszthető és a későbbiekben a szórás szintén megadható, mint optimalizálandó mennyiség, ezáltal még jobb egyezés is elérhető lenne. Az eset jól demonstrálja, hogy a saját készítésű GPU alapú DEM szoftverrel akár egy nap alatt lehetséges a paraméterek kalibrálása és a mérés reprodukálása, amely meglehetősen rövidnek számít populáció alapú algoritmussal történő DEM paraméterkalibrációk esetén.

4. KONKLÚZIÓ

Bemutatásra került, hogy a saját fejlesztésű GPU alapú DEM megoldó implementálása sikeresen megtörtént, amely használható akár talaj-kultivátorszerszám egymásra hatásának a szimulációjára. A fő eredmény, hogy a saját fejlesztésű megoldó a tízedére csökkenti a futási időt más szoftverekhez képest, amely elsősorban GPU-s gyorsításnak köszönhető. Más GPU-s DEM szoftverekkel ellentétben a saját szoftverünk általánosan használható, bármilyen problémára, illetve több különböző méretű és anyagú szemcsét is alkalmazni lehet. Az alacsony futási idő egyrészt lehetővé teszi sok szemcse alkalmazását. Másrészt az alacsony futási időnek köszönhetően nagy számú szimuláció futtatható viszonylag rövid idő alatt, és ez lehetővé teszi a numerikus optimalizációt is, amely segítségével sikerült reprodukálni egy homok talajban készült vonóerő mérést. Itt csak egy egyszerű példa került bemutatásra, de természetesen több mérésre együttesen is lehetséges a paraméteroptimalizáció, a fitness függvény megfelelő definiálásával. A saját fejlesztésű program hatékonyságát jól demonstrálja, hogy az optimalizáció futási ideje kevesebb volt mint egy nap, pedig 2700 ülepítés és kultivátoros talajművelés történt meg több mint 30 000 szemcsével minden szimulációban.

KÖSZÖNETNYILVÁNÍTÁS

A Kulturális és Innovációs és Minisztérium **ÚNKP-23-3-I-BME-51** kódszámú Új Nemzeti Kiválóság Programjának a Nemzeti Kutatási, Fejlesztési és Innovációs Alapból finanszírozott szakmai támogatásával készült. A Doktoranduszi Kiválósági Ösztöndíj Program (DKÖP) által támogatott projekt a Kulturális és Innovációs Minisztérium Nemzeti Kutatási Fejlesztési és Innovációs Alapból nyújtott támogatásával, a Nemzeti Kutatási, Fejlesztési és Innovációs Hivatallal kötött támogatási szerződés alapján valósult meg. A dolgozat a Magyar Tudományos Akadémia Bolyai János Kutatási Ösztöndíjának támogatásával valósult meg. Ez a kutatás a TKP-6-6/PALY-2021 számú projekt a Kulturális és Innovációs Minisztérium Nemzeti Kutatási Fejlesztési és Innovációs Alapból nyújtott támogatásával, a TKP2021-NVA pályázati program finanszírozásában valósult meg. A kutatás az ÚNKP támogatásával, a Nemzeti Kutatási, Fejlesztési és Innovációs Alap támogatásával, ÚNKP-23-5-BME-80 pályázati számon valósult meg. A dolgozat az Országos Tudományos Kutatási Alap (NKFIH FK-146067) támogatásával valósult meg.

IRODALMI HIVATKOZÁSOK

- [1] Rádics J.P., Jóri I.J., *Development of 3E tillage system and machinery to challenge climate change impacts*, Periodica Polytechnica Mechanical Engineering, Faculty of Mechanical Engineering of the Budapest University of Technology and Economics, 2010, Vol. 54, 49-56
- [2] Z. Asaf, D. Rubinstein, I. Shmulevic., *Evaluation of link-track performances using DEM*, Journal of terramechanics, Elsevir, 2006, Vol. 43, 141-161
- [3] Tamás K., Jóri I.J., Mouazen A.M., *Modelling soil-sweep interaction with discrete element method*, Soil and Tillage Research, Elsevir, 2013, Vol. 134, 223-231
- [4] R. Cai és mtsai, *Modified cell-linked list method using dynamic mesh for discrete element method*, Powder Technology, Elsevir, 2018, Vol. 340, 321-330
- [5] Pásthly L., Gräff J., Tamás K., *Development Of A 2D Discrete Element Software With LabVIEW For Contact Model Improvement And Educational Purposes*, ECMS, 2022, 203-209
- [6] Feoktistov V., *Differential evolution*, Springer, 2006
- [7] Pásthly L., Farkas Zs., Tamás K., *Laboratóriumi talajvályús mérőberendezés fejlesztése*, Mezőgazdasági Technika LXIV, 2023, 2-7