

Útvonaltervezés heurisztikus módszerekkel – a hangyakolónia algoritmus

Route planning with heuristic methods - the ant colony algorithm

Polyák Gabriella*, Póth Miklós**

* Szabadkai Műszaki Szakfőiskola, Szabadka, Szerbia

** Szabadkai Műszaki Szakfőiskola, Szabadka, Szerbia

*poljak.gabriela@vts.su.ac.rs

**pmiki@vts.su.ac.rs

Összefoglalás — A munka célja egy útvonaltervezési probléma megoldása heurisztikus módszerrel. A problémát lehet az utazó ügynök problémához hasonítani, ugyanis útvonaltervezésről, illetve útvonal optimalizációról van szó. A munka alapját a Formula-1-es versenynaptár nagydíjhelyszíneinek földrajzi koordinátái képezik, azonban bármilyen más, nyilvánosan elérhető adathalmazon is futtatni lehet a programokat. A munkában a Microsoft Excel egyik bővítménye, az Excel Solver lett használva, illetve egy MATLAB-ban íródott program, ami a hangyakolónia algoritmust alkalmazza. A hangyakolónia algoritmus azon a természetbeni megfigyelésen alapul, hogy minden hangya nyomot hagy maga után, egy bizonyos feromon nevű vegyi anyagot, és minél több hangya követi ugyanazt az utat, annál több a lesz a feromon, és ez minden következő hangyának "pozitív információ" az adott út helyességéről. A munkában az Excelben és a MATLAB-ban kapott eredmények kerülnek összehasonlításra és összegzésre.

Kulcsszavak: útvonaltervezés, Excel Solver, hangyakolónia algoritmus, Formula-1

Abstract — The aim of this paper is to solve a route planning problem using a heuristic method. The problem can be compared to the traveling salesman problem, because it is about route planning and route optimization. The basis of the work is the geographical coordinates of the grand prix locations of the Formula 1 race calendar, but the programs can also be run on any other publicly available data set. An extension of Microsoft Excel, Excel Solver, was used in the paper, as well as a program written in MATLAB, which uses the ant colony algorithm. The ant colony algorithm is based on the natural observation that each ant leaves a trail, a certain chemical substance called a pheromone, and the more ants follow the same path, the more pheromone will be, and this is "positive information" for each subsequent ant. about the correctness of a given road. In this work, the results obtained in Excel and MATLAB are compared and summarized.

Keywords: route planning, Excel Solver, ant colony algorithm, Formula 1

1 BEVEZETÉS

A munka alapelvei az utazó ügynök probléma elvein alapulnak, mivel útvonaltervezésről, illetve annak optimalizációjáról van szó. Az utazó ügynök probléma esetén az ügynöknek n számú várost kell bejárnia, hogy az utazási költségek minimálisak legyenek. Ennek a problémának két változata van. Az első változatban az ügynöknek vissza kell térnie a kiindulási helyzetbe, így a probléma megoldása a Hamilton-kör. A második változatban az utazó ügynök kezdő és záró városa eltérő, így ebben az esetben a megoldás Hamilton-út [1].

A közelmúlt világméretű problémái, mint a koronavírus-járvány és Kelet-Európa bonyolult geopolitikai helyzete, illetve a jelenlegi energiaválság, jelentős hatással voltak és vannak az emberek mindennapi életének minden szegmensére, így a sportra, pontosabban az autó- és motorsportra is. Az autó- és motorsport sorozatok csúcskategóriái folyamatosan szigorúbb energiafogyasztási korlátozásokat vezetnek be, hogy ösztönözzék a nagy hatékonyságú hajtáslánc-technológia fejlesztését. A Formula-1-ben a verseny során korlátozott az üzemanyag fogyasztás, valamint korlátozott a hibrid rendszer villamosenergia felhasználása [2].

Az üzemanyag fogyasztás és az üvegház hatású gázok kibocsátásának csökkentése érdekében rendkívül fontos lenne a Formula-1-es versenynaptár optimalizálása. Ebben az esetben a versenyautók, a felszerelések, az alkatrészek és a csapatok éves szinten sokkal kevesebbet utaznának, ez a megtakarítás pedig több ezer kilométerben mérhető. Az üvegház hatású gázok kibocsátása és az üzemanyag fogyasztás csökkenése mellett jelentős anyagi és időmegtakarítás is elérhető.

2 IRODALMI ÁTTEKINTÉS

Az utazó ügynök probléma (TSP) az egyik legszélesebb körben tanulmányozott kombinatorikus optimalizálási probléma. Állítása meglehetősen egyszerű, mégis továbbra is az operációkutatás egyik legnagyobb kihívást jelentő problémája. A TSP leggyakoribb gyakorlati értelmezése az, hogy egy utazó ügynök a legrövidebb utat keresi n ügyfelen vagy városon keresztül. Ez az alapvető probléma számos járműirányítási alkalmazás alapját képezi [3].

Amikor a gyakorlatban optimalizálni kell egy utazó ügynök problémát, gyorsan számos nehézségbe előtérbe kerülhet. Előfordulhat, hogy nincs kéznél algoritmus egy példa optimális megoldására, és hogy az idő vagy a tudás nem teszi lehetővé egy ilyen algoritmus fejlesztését és megvalósítását. A problémák egyszerűen túl nagyok lehetnek, és ezért meghaladják még a legjobb algoritmusok képességeit is az optimális megoldások megtalálásához. Másrészt az is előfordulhat, hogy a számításához szükséges idő nem elegendő ahhoz, hogy egy algoritmus elérje az optimális megoldást. Mindezekben az esetekben határozottan szükség van olyan közelítő (heurisztikus) algoritmusokra, amelyek jó minőségű megoldásokat határoznak meg, és az adott korlátok mellett a legjobb elérhető eredményeket adják [4].

Számos heurisztikus technikát alkalmaztak a probléma hatékony megoldásának megtalálására, mint például a hangya kolónia algoritmusok, szimulált hűtés, részecske-raj alapú optimalizálás és genetikai algoritmusok [5]. De ahogy a városok száma növekszik, a megoldás megtalálásához szükséges számítások nehézkessé válnak. A számítási nehézségek ellenére olyan módszereket lehet használni, mint a genetikai algoritmusok és a tabu keresés, amelyek közel optimális megoldást nyújthatnak több ezer városra is [6].

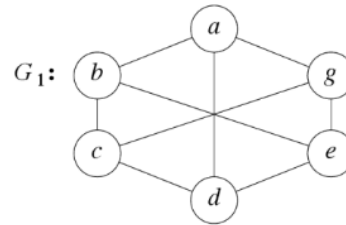
A hangya kolónia algoritmus az utazó ügynök probléma megoldásának egyik metaheurisztikus módszere. A hangya kolónia algoritmus hasonlít a valódi hangyák viselkedésére. Miután élelmet találtak, visszatérnek a kolóniába, miközben feromonnyomokat hagynak maguk után. Ha más hangyák találnak ilyen utat, valószínűleg nem véletlenszerűen választanak más utat, hanem inkább követik a nyomot. Nagyobb a valószínűsége annak, hogy a hangyák a magasabb feromonkoncentrációjú nyomvonalat választják. A feromonnyom lehetővé teszi a hangyák számára, hogy visszatérjenek az élelmiszerforráshoz, és fordítva. A nyomvonalat más hangyák is használják a felfedezett táplálékforrás megtalálására. Ahogy egyre több hangya választja ki a követendő utat, az ösvényen lévő feromon felhalmozódik, vonzóbbá téve azt más hangyák számára is [6].

3 ELMÉLETI MEGALAPOZÁS

Ebben a fejezetben a munka alapelemei, az útvonaltervezési módszerek és kritériumok kerülnek bemutatásra. A munkában a Microsoft Excel egyik bővítménye, az Excel Solver lett alkalmazva, illetve a MATLAB-ban egy hangyakolónia algoritmust alkalmazó program.

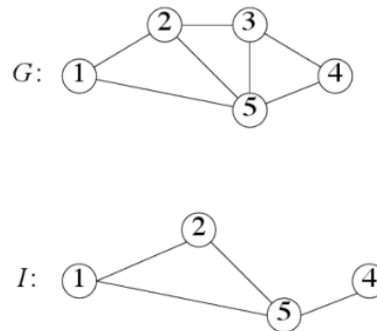
3.1 Gráfok

A gráfok az alapvető elemek ebben a munkában, mivel a helyek matematikai absztrakcióját reprezentálják, a köztük lévő élek pedig a helyek közötti útvonalakat. Számos probléma megoldására szolgálnak, amelyek valós vagy absztrakt objektumok formájában és valós vagy absztrakt kapcsolatok formájában jeleníthetők meg ezek között az objektumok között, például: forgalom, távközlési hálózat stb. A G_1 gráf két véges halmazból áll: a csomópontok összességéből (N) és a köztük lévő élek összességéből (C) úgy, hogy minden $g \in C$ kapcsolat két i és j csomópontot köt össze (1. ábra).



1. ábra: G_1 gráf, példa [7]

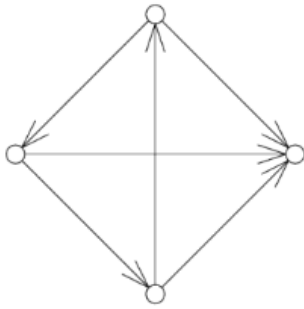
Egy egyszerű G gráf az, amelyben nincsenek párhuzamos élek (két azonos csomópontot összekötő kapcsolatok) és ciklikus élek (egy csomópontot önmagával összekötő kapcsolatok). Egy egyszerű gráfban az élek legnagyobb száma egyenlő az $NC(NC-1)/2$ -vel, ahol NC az N halmaz csomópontjainak száma. A teljes gráf egy olyan egyszerű gráf, amely az élek maximális számát tartalmazza. Ha két csomópont között összekötő él van, akkor azok szomszédos csomópontok. A G gráf részgráfja bizonyos csomópontok és a hozzájuk tartozó élek részhalmaza (2. ábra).



2. ábra: G gráf és I részgráf [7]

Ha a gráf csomópontjai konkrét objektumok, amelyek meghatározott pozícióval rendelkeznek a térben, akkor ezek euklideszi gráfok. Az útvonal az N -ből származó csomópontok sorozata, ahol minden két szomszédos csomópontot egy C -ből származó él köt össze. Ha az út ugyanabban a csomópontban kezdődik és végződik, akkor ez egy gyűrű (ciklus vagy ciklikus út). Az egyszerű útvonal olyan sorozat, amelyben a csomópontok nem ismétlődnek. Az Euler-útvonal (ciklus) olyan útvonal (ciklus), amely az összes élet tartalmazza, és amelyben minden él csak egyszer jelenik meg [8].

Egy gráf akkor kapcsolódik össze, ha van legalább egy él, amely bármely két csomópontot összeköti. Az irányított gráf olyan gráf, amelyen egy összefüggés rendezett pontpárként jelenik meg, és $(i, j) \neq (j, i)$. A digráfban különböző orientációjú párhuzamos kapcsolatok megengedettek (3. ábra).



3. ábra: Digráf, példa [7]

Gráfelméleti algoritmusok:

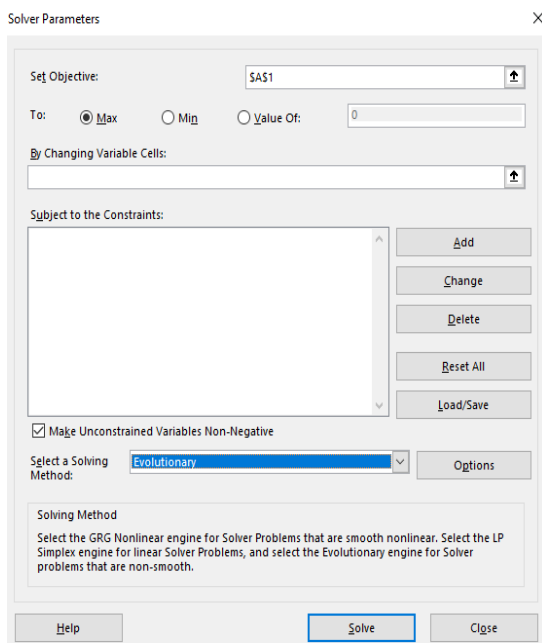
- Szélességi keresés (Breadth First Search – BFS)
- A legrövidebb út megtalálása – Bellman optimalitási elve
- Mélységi keresés (Depth First Search - DFS)

3.2 Útvonaltervezési módszerek

Ebben a részben a munkában használt módszerek, azaz az Excel Solver és a hangyakolónia algoritmus kerülnek bemutatásra.

3.2.1 Excel Solver

A Solver programok olyan matematikai optimalizáló alkalmazások, amelyek matematikai problémák megoldására használhatók. A cél egy olyan általános modell megalkotása, amelynek segítségével sok hasonló probléma megoldható. Az Excel Solver működési elve, hogy a korlátok figyelembevételével megvizsgálja a lehetséges megoldásokat, majd kiválasztja közülük a legjobbat. Az Excel Solver segítségével megkereshető egy képlet optimális értéke (minimális, maximális vagy specifikus célérték) egy úgynevezett céllében. Az Excel Solver ablaka a 4. ábrán látható.



4. ábra: Az Excel Solver ablakának kinézete

Az út létrehozása a (1) képlettel történik, ahol a valószínűsége, hogy a k . hangya az i . pontból a j . pontba megy az algoritmus t . iterációjában, az alábbi [10]:

$$p_{ij}^k = \frac{[\tau_{ij}(t)]^{\alpha} * [\eta_{ij}]^{\beta}}{\sum_{l \in N_i^k} [\tau_{il}(t)]^{\alpha} * [\eta_{il}]^{\beta}}, \text{ ha } j \in N_i^k \quad (1)$$

Ahol $\eta_{ij} = \frac{1}{d_{ij}}$, két csomópont közötti távolság reciprok értéke, $\tau_{ij}(t)$ pedig az (i, j) él feromontartalma.

A feromon frissítésére a (2) képlet kerül alkalmazásra:

$$\tau_{ij}(t + 1) = (1 - \rho) * \tau_{ij}(t) + \sum_{k=1}^m \Delta \tau_{ij}^k(t) \quad (2)$$

Ahol

$$\Delta \tau_{ij}^k(t) = \begin{cases} \frac{1}{L^k(t)}, & \text{ha az } (i, j) \text{ élen átment a } k. \text{ hangya} \\ 0, & \text{különben} \end{cases} \quad (3)$$

3.3 Útvonaltervezési kritériumok

Az útvonaltervezés kritériumai gyakran a távolságon vagy az utazási időn alapulnak [11]. A versenynaptár optimalizálásakor több szempontot is figyelembe kell venni, hiszen a versenyek sorrendjének meghatározása és a közöttük lévő útvonal tervezése összetett probléma. A versenynaptár útvonalát a következő kritériumok szerint lehet tervezni:

- Legrövidebb útvonal
- Leggyorsabb útvonal
- Időjárás és éghajlat
- Pénz
- Politikai és vallási aktualitások
- Egyéb hatások

3.3.1 Legrövidebb útvonal

Ebben a munkában az útvonal tervezése csak ez a kritérium alapján történt. Ha ez alapján tervezzük a versenynaptárt, az azt jelenti, hogy az autóknak, felszereléseknek, alkatrészeknek és a csapatoknak több ezer kilométerrel kevesebbet kell megtenniük, ezzel csökkenne az üzemanyag fogyasztás és a káros gázok kibocsátása is, illetve jelentős pénzügyi és időbeli megtakarítás is lenne.

3.3.2 Leggyorsabb útvonal

Gyakran nem a legrövidebb út a leggyorsabb is, ennek oka lehet a sebességkorlátozás, a terep jellege vagy a forgalmi torlódás.

3.3.3 Időjárás és éghajlat

Egy adott év Formula-1-es szezonja általában márciusban kezdődik. Mivel ilyenkor Európában általában még elég hűvös van, így az időjárás nem megfelelő a versenyzésre. Az elmúlt jó pár évben a szezonkezdet vagy valamelyik közel-keleti versenyhelyszínen volt, vagy Ausztráliában. Az első 2-3 megrendezett futam után, mikor Európában is emelkedni kezd a napi átlaghőmérséklet, akkor következnek az európai versenyek és kisebb-nagyobb kivételekkel a szárgüldő cirkusz őszi leginkább az öreg kontinensen is marad. Ősszel az időjárás körülmények miatt következnek a

távol-keleti helyszínek, majd Dél-Amerika és végül ismét a Közel-Kelet.

3.3.4 Pénz

A versenynaptár nagydíjainak sorrendjében az időjárás mellett a pénzé a legfontosabb szerep. A naptárban az első és az utolsó nagydíj lebonyolításáért plusz díjat kell fizetni, amit a legtöbb ország nem tud kifizetni, így hosszú évek óta az év utolsó nagydíja az Abu Dzabi Nagydíj. A pálya is a legrágább a Forma-1 eddigi történelmében, mivel a teljes komplexum megépítése több mint 1 milliárd dollárba került [12]. A nagydíjak szervezői átlagosan körülbelül 40 millió dollárt fizetnek a Liberty Media Corporationnek, a Formula-1 többségi tulajdonosának egy nagydíj megszervezésének lehetőségéért [13].

3.3.5 Politikai és vallási aktualitások

Az Orosz Nagydíjat 2022-ben a Kelet-Európában jelen levő bonyolult geopolitikai helyzet miatt azonnal törölték a versenynaptárból, így a naptár 22 nagydíjhelyszínt tartalmazott. A 2024-es versenynaptár szempontjából pedig fontos a ramadán kezdeti dátuma, ugyanis március elejére esik, abban az időszakban pedig akkor nem rendezhetnek nagydíjat a közel-keleti, főleg iszlám vallásúak lakta országokban, a ramadán végeztével, ami április elejére esik, pedig már túl magas a hőmérséklet egy nagydíjhétvége megszervezéséhez abban a térségben.

3.3.6 Egyéb hatások

A korábbi években különösen nehéz volt a Formula-1-es versenynaptár megszervezése. 2020 letelején az Ausztrál Nagydíjat törölték az ausztrál erdőtüzek miatt. A következő futamnak és a szezon hivatalos rajtjának Bahreinben kellett volna lennie, ahová minden csapat meg is érkezett, azonban pénteken, egy órával az első szabadedzés kezdete előtt, a teljes versenyhétvégét törölték a koronavírus-járvány miatt. Közvetlenül ezután a többi futamot is törölték, így a 2020-as szezon hivatalos rajtja július elején volt Ausztriában. 2020-ban 14 helyszínen rendeztek, összesen 17 nagydíjat. 2021-ben a járvány miatt szintén történtek változások, de végül megtartották a nagydíjakat; 21 helyszínen és összesen 22 nagydíjat. A 2022-es évre visszatértek azok a nagydíjak, amelyeket a járvány miatt az előző két évben nem rendeztek meg, azaz az Ausztrál, a Kanadai, a Japán és Szingapúri Nagydíj. A Kínai Nagydíj a járvány miatt talán majd csak 2024-ben kerül vissza a versenynaptárba. A Katari Nagydíj 2021-ben debütált, de 2022-ben a labdarúgó-világbajnokság miatt nem került megrendezésre.

4 GYAKORLATI MEGVALÓSÍTÁS – ÚTVONALTERVEZÉS

Az útvonaltervezés többféle módon történt. Először Microsoft Excelben, az Excel Solver nevű bővítményével, majd a MATLAB-ban írt hangyakolónia algoritmust használó programban.

4.1 Excel Solver

A 2020-as versenynaptár végül 14 nagydíjhelyszínből és 17 megrendezett nagydíjhétvégéből állt. Három alkalommal dupla hétvége volt, ami azt jelenti, hogy két egymást követő hétvégén ugyanazon a helyszínen két, különböző nevű nagydíj került megrendezésre (1. táblázat).

1. táblázat: A versenypályák elnevezései és a földrajzi koordinátáik (2020)

	Versenypálya	Földrajzi szélesség	Földrajzi hosszúság
1	Red Bull Ring	47,219722	14,764722
2	Hungaroring	47,582222	19,251111
3	Silverstone Circuit	52,078611	-1,016944
4	Circuit de Barcelona-Catalunya	41,57	2,261111
5	Circuit de Spa-Francorchamps	50,437222	5,971389
6	Autodromo Nazionale Monza	45,620556	9,289444
7	Autodromo Internazionale del Mugello	43,9975	11,371944
8	Sochi Autodrom	43,410278	39,968271
9	Nürburgring	50,335556	6,9475
10	Autódromo Internacional do Algarve	37,232	-8,632
11	Autodromo Internazionale Enzo e Dino Ferrari	44,341111	11,713333
12	Intercity Istanbul Park	40,951667	29,405
13	Bahrain International Circuit	26,0325	50,510556
14	Yas Marina Circuit	24,467222	54,603056

A nagydíjhelyszínek közötti távolságot a (4) képlet alapján számoljuk, ami az euklideszi távolság kiszámításának képlete.

$$d(p, q) = \sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2} \quad (4)$$

A 2. táblázatban a bal oldalon az optimalizálás előtti euklideszi módszerrel kiszámított távolságok, a sorrend és a távolságok összege látható, ez az a sorrend, ami a valóságban is volt, tehát a versenynaptár sorrendje. A jobb oldalon az optimalizálás utáni eredmények láthatók, az optimalizált út körülbelül 1/3-dal rövidült. Az eredeti összeg 252,741 (26 035 km), míg optimalizálás után 154,036 (15 867 km) lett az eredmény.

2. táblázat: Az útvonal hossza optimalizálás előtt és után (2020)

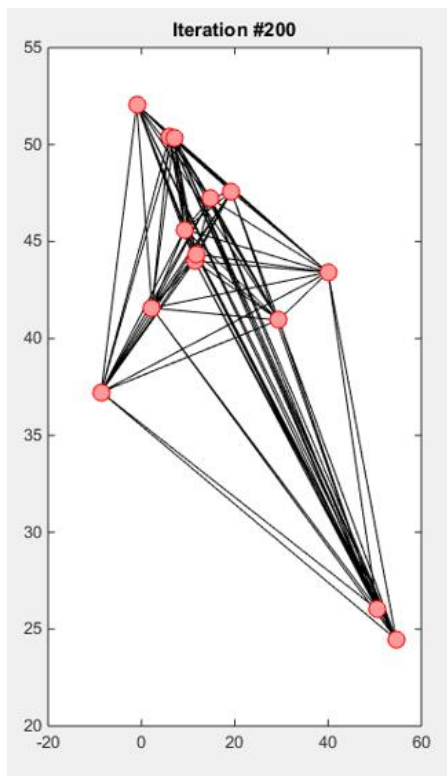
Sorszám	Távolság	Sorszám	Távolság
1	4,50101	1	8,41531
2	20,7608	9	0,98139
3	11,008	5	7,17851
4	9,61217	3	16,6857
5	5,84891	10	11,7251
6	2,64029	4	8,112
7	28,6024	6	2,64029
8	33,7392	7	0,48437
9	20,3574	11	18,0134
10	21,5516	12	25,8462
11	18,0134	13	4,38163
12	25,8462	14	23,9378
13	4,38163	8	21,1331
14	45,8778	2	4,50101
1		1	

Összesen	252,741	Összesen	154,036
----------	---------	----------	---------

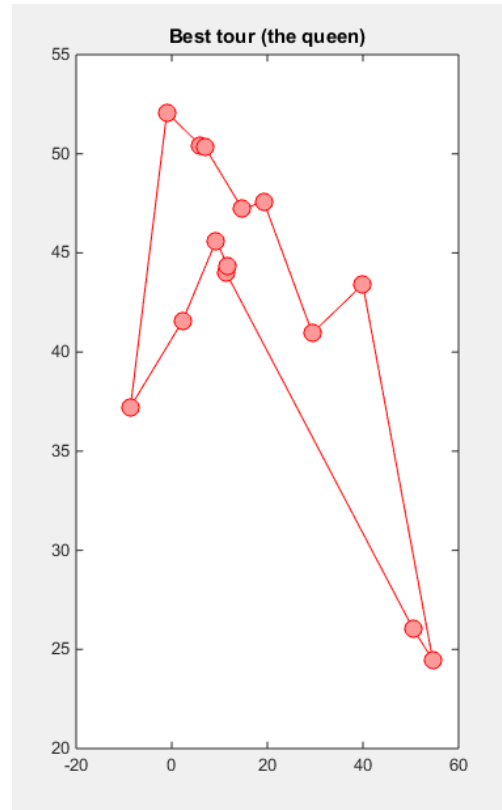
A 2020-as versenynaptár után ugyanez a módszer lett alkalmazva a 2021-es, 2022-es és 2023-as versenynaptárra, valamint az eddigi összes, azaz 77 nagydíjhelyszínre.

4.2 MATLAB – hangyakolónia algoritmus

A hangyakolónia algoritmust alkalmazó program tetszőleges számú hangyára és iterációra futtatható. A 2020-as versenynaptár 14 nagydíjhelyszíne esetén 10 hangyára és 200 iterációra lett lefuttatva a program. Az 5. ábrán látható a 200. iterációban a 10 hangya által bejárt útvonalak. A program lefutása után a legrövidebb útvonal hosszúsága 155,181 (15 985 km). A 6. ábrán koordináta-rendszerben ábrázolva látható a kapott legrövidebb útvonal. A 2020-as versenynaptár után ugyanez a módszer alkalmazva lett a 2021-es, 2022-es és 2023-as versenynaptárra, valamint az eddigi összes, azaz 77 nagydíjhelyszínre.



5. ábra: A 200. iterációban a 10 hangya által bejárt útvonalak



6. ábra: A kapott legrövidebb útvonal koordináta-rendszerben ábrázolva

5 EREDMÉNYEK, ÖSSZEZÉS

A 3. táblázatban összegezve láthatóak a kapott legfontosabb eredmények. A táblázat az összes módszert (Solver és hangyakolónia algoritmus) tartalmazza, illetve az összes esetet amire alkalmazva lettek (2020-as, 2021-es, 2022-es, 2023-as versenynaptár és 77 nagydíjhelyszín). A 2020-as versenynaptár esetén az eredeti útvonal közel 40%-kal csökkent, a Solverrel és a hangyakolónia algoritmussal kapott eredmények között kicsit több mint 100 km a különbség. A 2021-es versenynaptár esetén szintén közel 40 %-os javulás látható, ebben az esetben közel 1000 km-re rövidebb útvonal lett meghatározva a hangyakolónia algoritmus segítségével, mint Excel Solverben. A 2022-es versenynaptár esetében az útvonal az eredetihez képest közel 55%-kal csökkent, közel 2000 km különbség van a Solver és a hangyakolónia algoritmus között. A 2023-as versenynaptár esetében az útvonal kevesebb mint felére csökkent az eredetihez képest, az Excel Solver jobban teljesített, mint a hangyakolónia algoritmus, ugyanis több mint 3000 km-rel rövidebb lett a kapott útvonal. Az összes eddigi, azaz 77 nagydíjhelyszín esetében lényegesen nagyobb a két módszer között különbség, mint az előző esetekben. A hangyakolónia algoritmus segítségével kapott útvonal közel 45 000 km-rel rövidebb, mint az Excel Solverben kapott. Nagyobb mennyiségű bemeneti adattal az Excel Solver nehezebben dolgozik, míg a kis mennyiségű adatokkal nincs problémája. A hangyakolónia algoritmus pedig kiválóan boldogul nagyobb mennyiségű bemeneti adattal is, csak a program futásának ideje lesz hosszabb.

3. táblázat: Összefoglaló táblázat a kapott eredményekről

	Eredeti	Excel Solver	Hangyakolónia algoritmus
2020	26 035 km	15 867 km	15 985 km
2021	64 947 km	40 823 km	39 684 km
2022	139 875 km	66 304 km	68 259 km
2023	149 087 km	69 484 km	72 746 km
77 helyszín	562 456 km	136 599 km	91 782 km

A jövőbeni tervek között szerepel még több metaheurisztikus módszer alkalmazása, illetve az útvonaltervezés végrehajtása a munkában említett többi kritérium szerint is.

IRODALOMJEGYZÉK

- [1] Stankovic, V. & Jovic, J (2006). *Ekonomika – Problem trgovačkog putnika*. vol. 52, br. 5-6, str. 198-202
- [2] Liu, X., Fotouhi, A. & Auger, D. (2022). Application of advanced tree search and proximal policy optimization on formula-E race strategy development. DOI: <https://doi.org/10.1016/j.eswa.2022.116718>
- [3] Laporte, G. (1992). The traveling salesman problem: An overview of exact and approximate algorithms. *European Journal of Operational Research*, 59(2), 231–247. doi:10.1016/0377-2217(92)90138-y
- [4] Jünger, M., Reinelt, G., & Rinaldi, G. (1995). Chapter 4 The traveling salesman problem. *Network Models*, 225–330. doi:10.1016/s0927-0507(05)80121-5
- [5] Kylie Bryant, Arthur Benjamin, Advisor, “Genetic Algorithms and the Traveling Salesman Problem”, Department of Mathematics, December 2000.
- [6] Dahiya C. & Sangwan S. (2018). Literature Review on Travelling Salesman Problem, *International Journal of Research - June 2018*
- [7] Gould, R. *Graph Theory*. Courier Corporation, 2012 ISBN: 9780486498065
- [8] Dasic T., & Stanic M. (2014). *Metode optimizacije – Grafovi algoritmi*
- [9] Dorigo, M., Birattari, M., & Stutzle, T. (2006). Ant colony optimization. *IEEE Computational Intelligence Magazine*, 1(4), 28–39. doi:10.1109/mci.2006.329691
- [10] Agárdi, A. Klaszterezés hangya kolónia optimalizációval és tabu kereséssel. *Multidiszciplináris tudományok*, 11. kötet. (2021) 4 sz. pp. 120-126. <https://doi.org/10.35925/j.multi.2021.4.14>
- [11] Sampaio, C., Bandeira, J. M., Macedo, E., Vilaça, M., Guarnaccia, C., Friedrich, B., Coelho, M. C. (2019). A Dynamic Link-based Eco-indicator for supporting equitable traffic management strategies. *Transportation Research Procedia*, 37, 43–50. doi:10.1016/j.trpro.2018.12.164
- [12] <https://wtf1.com/post/how-much-does-it-cost-to-host-an-f1-race/>
- [13] <https://hu.motorsport.com/fl1biz/news/mennyibe-kerulhet-egy-formula-1-es-auto/4392635/>