

Absztrakt

Az önvezető járművek, különösen a drónok a közeljövőben nagy eséllyel könnyen fogják tudni csökkenteni a kiszállítást végző futár cégek logisztikai költségeit. A drónokkal való kiszállítás azonban számos kérdést felvet, amelyekkel ez a projekt munka foglalkozik. A kutatás célja kettős volt: alkotni egy szimulációt, mely felhasználható arra, hogy légi járművekkel modellezhessük ételrendelések kiszállítását. A szimuláció paraméterként megkapja az éttermek és vásárlók helyzetét, generálja a megrendeléseket, illetve szimulálja a megrendeléseket kiszállító drónok működését (szél, sebesség, akkumulátor töltöttség, stb. figyelembevételével). További feladat egy ehhez felhasználható algoritmus kifejlesztése, mely optimalizálja a hálózat működését, a rendelési adatok alapján megadja az optimális drón-feladat párosítást.

Kulcsszavak

Szimuláció, Optimalizáció, Genetikus algoritmus, Brute-force

Motiváció

A minket körülvevő étel- és csomagkiszállító szolgáltatásokat nap mint nap használjuk. Az ezeket működtető cégek már egy ideje jelentős szerepet töltenek be a nagyvárosok környezetében azáltal, hogy mind a belvárosokban, mind azok vonzáskörzetében egyre inkább összehangolva, gyorsan és hatékonyan igyekeznek csomagot vagy ételt eljuttatni a lakosságnak. Nagymennyiségű kiszállítás során viszont már szükség van egy optimalizációra, amelyik meghatározza az optimális feladat hozzárendelést több drón esetén.

Hasonló fejlesztések

A dinamikus VRP-t kezelő szekvenciális stratégiák közé sorolhatjuk a következőket:

- **First Come First Served:** Az algoritmus egy FIFO sort használ, melynél minden egyes új elkészült, vagy elkészülendő rendelés a sor végéhez adódik hozzá.
- **Stochastic Queue Median:** Ennél a módszernél az adott jármű a számára előírt kiszolgálási helyszín mediánközpontján várakozik arra, hogy a rendszer igénybe vegye.
- **Nearest Neighbor:** A megoldás lényege, hogy amikor a munkavégző egység végez valahol a feladatával, a hozzá legközelebb eső feladathoz siet további munkavégzésre.
- **Shortest Job First:** A Shortest Job First algoritmus azt nézi meg, hogy az elkészült rendelések közül melyiket kell a legközelebbre kiszállítani az étteremből, majd azt osztja ki először a futárok közül.
- **Least Lost Value:** Az irányelv lényege, hogy a feladathoz értéket rendel hozzá, mely érték egy monoton csökkenő függvénnyel határozható meg.

Sikerek

- Kari TDK 3. helyezett
- OTDK részvétel
- 1 konferenciatick

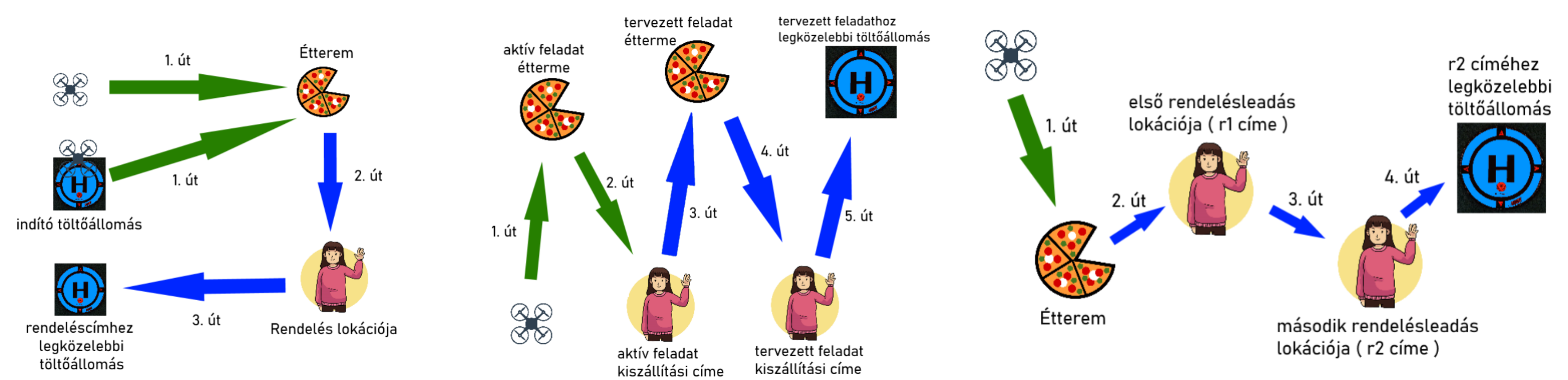
Saját módszer bemutatása

A rendeléskiosztás és ütemezés filozófiája a következő:

- egyszerre egy futár egy rendelést kaphat plusz feladatként,
- a kiosztás nem időablak szerint történik, hanem rendelésmennyiség szerint,
- futároktól nem veszünk el már rájuk osztott rendelést,
- bizonyos futár és rendelésmennyiség alatt nem indul el a rendeléskiosztás.

Az algoritmus az első lépése az, hogy sorrendet állít fel a kész vagy készülő rendelések között, idő által adott fontosságuk szerint. A lista első elemei, sorba rendezve, azok a rendelések lesznek, amik már készen vannak és kiszállításuk azonnal elkezdhető lenne. Ezek után a rendelések után kerülnek be azok az elemek növekvő sorban rendezve, amik még nincsenek készen. Így egy lineáris fontosságot tudunk meghatározni a rendelések között az idő alapján

Az algoritmus következő lépése arra szolgál, hogy felmérje, mely drón-rendelés párosítások kivitelezhetetlenek, majd a kivitelezhetőket "megméri" az Útvonalszámítás algoritmussal. A harmadik bemutatni kívánt algoritmus azt a feladatot végzi, hogy kiszámolja, ha az adott információk alapján elindítanánk egy drónt egy rendelésért, az milyen energiavesztéssel járna a drón számára és mennyi idő alatt sikerülne a rendelést kiszállítani



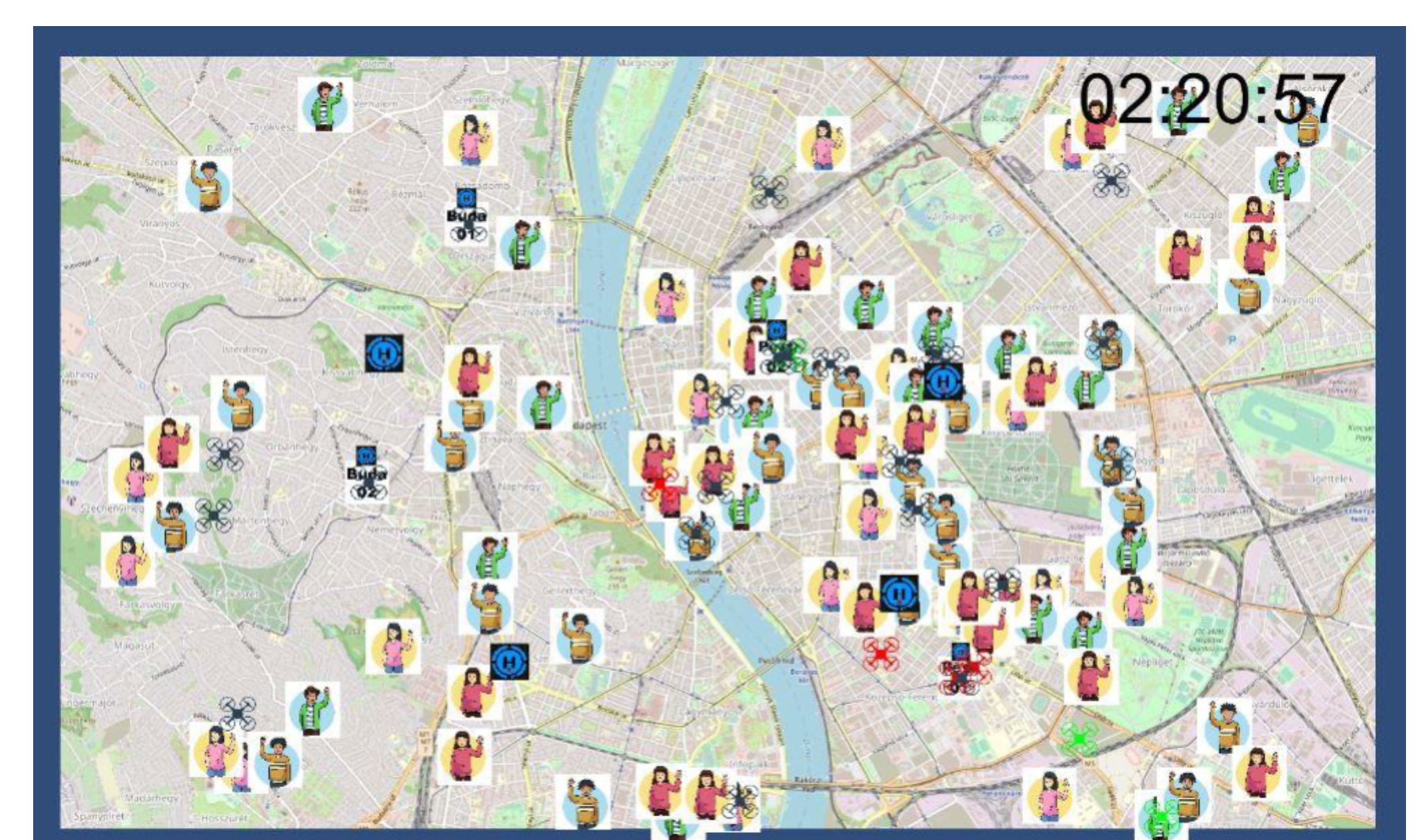
Megadott számú étterem, megadott számú megrendelésének kiszállításához megadott számú drón áll rendelkezésre. A megrendelés-drón párosításnak több megoldása is lehetséges, ezért egy optimalizációra van szükség az ideális kiosztás megállapításához. Az ehhez használt genetikus algoritmus működési terve az, hogy minden adott aktív rendelést megpróbál egy drónhoz társítani. A genetikus algoritmus egyedei tulajdonképpen ezek a lehetséges hozzárendelések. A fitness pedig figyelembe veszi, hogy egy adott kiosztás mennyire jó a gyakorlatban (várakozási idő, stb.). Kezdetben a kiosztásokhoz véletlenszerű párosítások generálódnak. Ezután megtörténik a fitness kiértékelése, mely alapján alkalmazódik az elitizmus, vagyis a legjobb értékű kiosztások egy része ezen a ponton elmentődik egy tömbbe.

Előfordulhat olyan eset is, hogy a körülményekhez képest viszonylag kevés rendelés generálódik, viszonylag kevés drónra, ezek ismétlés nélküli variáció verziókészletét tehát érdemes lehet brute force metódussal kiszámolni a genetikus algoritmusos megoldással szemben. Az itt implementált alapötlet számít arra a valóság alapú bekövetkezésre, hogy a rendszer bizonyos időn belül el fogja érni azt a rendelés- és drónmennyiséget, aminél már valószínűleg érdemes genetikus algoritmust alkalmazni. Az itt bemutatott algoritmus tehát várakozik, nem fut le addig, amíg bizonyos számú rendelés- és drón nem várakozik kiosztásra. Mindez egy konstanssal van limitálva, melyet futtatási minimumként nevezhetünk.

Eredmények értékelése

A két kitűzött célt sikerült elérni. A szimuláció képes olyan futtatásra, melyben a rendelések folyamatosan generálódtak, amelyeket aztán a drónok a céllokációkra vittek, így egyszerre megvalósítva akár 100 drón együttműködését és több, mint 1500 rendelés kiszállítását.

Elkészült egy egyszerű grafikus kezelőfelület, amelyen keresztül folyamatosan látható a szimuláció aktuális állapota, annak szereplői.



Az eredmények elemzése során számos vizsgálatra volt szükség, hogy validálni lehessen annak használhatóságát. Érdekes lehet a rendelések átlagos futásideje a drónok számának függvényében. Menet közben figyelni lehet a drónok kihasználtságát, töltöttségét és egyéb paramétereiket.

