Alkalmazott Matematikai Lapok **33** (2016), 93–118

AZ ÚTVONALTERVEZŐ ALGORITMUS TÖRTÉNETI ÁTTEKINTÉSE, KÜLÖNÖS TEKINTETTEL AZOK TURISZTIKAI CÉLÚ ALKALMAZÁSAIRA

APÁTHY M. SÁNDOR

Kevés rémesebb érzést tudok elképzelni, mint eltévedni egy idegen városban, főként, ha nyelvi akadályok miatt még segítségkérésre sem igen van lehetőségünk. Az idők során megannyi útvonaltervező eljárás született, és talán ezek születésében is hasonló élmények játszhattak közre. Cikkünk célja a jelenleg használatos turisztikai és szállítási célú útvonaltervező algoritmusokig vezető, helyenként rögös, és kitérőktől sem mentes tudományos út rövid bemutatása. A legrövidebb út problémájától indítva tudománytörténeti áttekintőnket láthatjuk, ahogyan a szerteágazó problémákat egyesíti a lineáris programozás technikája, majd rátérünk az utazóügynök feladatra, valamint az abból kifejlődő szállítási és útvonaltervezési problémákra koncentrálva. Mivel a 60-as évektől kezdve igen megnövekedett a különféle útvonaltervezési eljárások száma, így összefoglalónkat mindinkább a turisztikai megoldásokra szűkítjük, hogy tartani tudjuk a terjedelmi kereteket.

Kulcsszavak: Team Orienteering Problem, Route Planning, Heuristic Algorithm, Tourism JEL kód: C60, C61, Z32

1. Bevezetés

Az okostelefonok korában megannyi útvonaltervező alkalmazás áll rendelkezésünkre, hogy kirándulásaink megtervezésében segítségünkre legyen. Egy 3 napos út tervezése papíron, akár csak egy Budapest méretű városban is komoly kihívást jelent, és nem kevés munkaórát. Egy ilyen utat kevés személyes információ megosztása árán kalkulálni képes algoritmus, nyilvánvalóan temérdek tervezgetéssel töltött órát takaríthat meg a felhasználók számára, akik azon fáradoznak, hogy azt a néhány pihenésre szánt napot a számukra leginkább élvezetessé tegyék. Jelen cikk célja annak bemutatása, hogyan jutott idáig ez a tudományterület a legrövidebb út problémájától, mely már az őskorban is foglalkoztatta elődeinket az élelem lelőhely és a táboruk viszonylatában. Mint azt látni fogjuk, a lineáris programozási technika kifejlődése egységes keretet nyújtott a korábbi szerteágazó kombinatorikus optimalizálási feladatok megoldásának, melyeket korábban külön kezeltek. Ennek kialakulásában egy sor magyar tudós vett részt, akik munkásságának külön figyelmet szentelünk a cikkben. Az 50-es, 60-as évek megannyi olyan eredménnyel szolgált az operációkutatás területén, mely addig sohasem látott módon gyorsította fel a terület fejlődését. Hogy terjedelmi kereteinket tartani tudjuk, a korai eredmények ismertetése után az utazóügynök problémára fordítjuk figyelmünket, valamint az abból kifejlődő turisztikai és szállítmányozási célú eljárásokra. Az utolsó szakaszban a turisztikai útvonaltervező problémák néhány kiterjesztését mutatjuk be, illetve az azokra adott megoldásokat, számosságukra való tekintettel a teljesség igénye nélkül.

2. Legrövidebb út problémája

Az útvonaltervező algoritmusok szakirodalma messzire nyúlik vissza, hiszen már az őskorban is foglalkoztatta elődeinket, akárcsak az állatokat, hogyan tudnak a leggyorsabban, vagy leginkább energiatakarékos módon eljutni az élelem- vagy vízforráshoz. Első említést érdemlő mérföldköve a labirintusból való kijutást megoldó mélységi keresés algoritmusa (Depth-first search), mely Trémaux nevéhez fűződik. Az eljárás lényege, hogy adott pontból úgy járjunk végig egy gráfot, hogy addig megyünk előre a csomópontokon, míg lehetséges, majd visszalépünk az első olyan csomópontig, ahol elágazás volt, stb. Ez tehát egy mohó algoritmus, mely lokális optimumokon keresztül reméli elérni a globális optimumot. A mélységi bejárás módszeréről Wienernél olvashatunk először 1873-ban [96]. A legrövidebb utakra adott megoldások további történeti áttekintése előtt következzen egy definíció.

2.1. Definíció. (legrövidebb út irányítatlan gráfon) Legyen $G(V,E)$ irányítatlan gráf, V a csúcsok, E az élek halmaza, míg $P = (v_1, v_2, \dots, v_n) \in V \times V \times \dots \times V$ úgy, hogy v_i szomszédos v_{i+1}-gyel $\forall\ 1 \le i < n$, így P egy n hosszú út v_1 és v_n között. Legyen $e_{i,j}$ élköltség v_i és v_j csúcsok között, valamint az éleken legyen adott $f : E \to \mathbb{R}$ élköltség függvény. Ekkor v_0 és v^* között (ahol $v_0 = v_1$ és $v^* = v_n$) a legrövidebb út az, ami minden lehetséges n-re minimalizálja az alábbi kifejezést:

$$\sum_{i=1}^{n-1} f(e_{i,i+1}).$$

Az irányított gráfok esetén csupán annyi a különbség a definícióban, hogy irányított $e_{i,j}$ éleket követelünk meg a szomszédos v_i és v_j csúcsok között. Irányított gráfokra az 50-es években két megoldás is született. Ezen eljárásokban közös az alábbi, Ford [38] által leírt általános forma:

Legyen adott $G(V,E)$ irányított gráfon az $f : E \to \mathbb{R}$ élköltség függvény, és két csúcs közötti távolságot definiáló függvény, $d : E \times E \to \mathbb{R}$. Ekkor egy adott s csúcsból egy másik csúcsig tartó út hosszát az alábbiak szerint kalkuláljuk: legyen $d(s) = 0$ és $d(v_i) = \infty\ \forall\ v_i \in V/s$. Válasszuk $(v_j,\ v_i)$ élt, ahol $d(v_i) > d(v_j)+$

$+ f(v_j, v_i)$ és legyen $d(v_i) := d(v_j) + f(v_j, v_i)$, majd folytassuk ezt addig, amíg már nem találunk ilyen élt. A két módszer közötti különbség ott van, ahogyan az iterációban a következő élt kiválasztjuk:

- A Bellman-Ford algoritmusban minden iterációban végigmegyünk az éleken, míg el nem fogynak, összesen maximum $|V|$ darab iterációban. Ezt a módszert (vagy ezzel ekvivalens módszert) írt le egymástól függetlenül Shimbel 1955-ben [79], Bellman 1958-ban [5] és Moore 1959-ben [66]. Shimbel telefonhálózatok mátrix reprezentációján igyekezett megoldani a legrövidebb út problémját.
- A Dijkstra által 1959-ben közzétett algoritmusban [29] mindig a legkisebb $d(v_j)$ értékhez tartozó (v_j, v_i) élt választjuk, így minden él legalább egyszer kiválasztásra kerül, ha nincsenek negatív élköltségek. Ezzel ekvivalens megoldást írtak le Leyzorek et al. [62], a Case Institute of Technology kutatói is 1957-es riportjukban, és Shimbel korábbi eredményének komplexitásán is tudtak javítani javaslatukkal. Hasonló, és csak kicsit lassabb algoritmus az 1958-ban Dantzig cikkében megjelent módszer [25], amely szerint azt az élt kell választani a következő lépésben, amelyre a $d(v_j) + f(v_j, v_i)$ érték minimális.

Mint látható az évszámok közelségéből is, a korszak igen termékeny volt, a megoldások pedig kis túlzással egyszerűek, hiszen több kutató egymástól függetlenül is ekvivalens eredményre jutott. A korszakról bővebben Schrijver cikkében olvashatunk [78].

Rövid kitérő erejéig meg kell említenünk a témával kapcsolatosan a *minimális feszítőfa problémát*, mely egy összefüggő, irányítatlan gráfban a legkisebb összélköltségű feszítőfát keresi. (Feszítőfa alatt azt a fát értjük, amely a gráf összes csúcsát tartalmazza, élei a gráf eredeti élei, és minden csúcsból, minden csúcsba pontosan egy út vezet). A problémára már 1926-ban adott egy megoldást Boruvka [9], melynek egy egyszerűsített változatát írta meg Jarník 1929-es levelében Boruvkának, majd 1930-ban cseh nyelven cikk formájában is megjelent [53]. Ám ez feledésbe merült, és tőle függetlenül Prim 1957-ben [71], valamint Dijkstra 1959-ben ismét megalkották az eljárást [29], ezzel sikerült javítaniuk Kruskal 1956-ban megjelent megoldásának számításigényén [60], melyet Boruvka nyomán írt. Az eljárás igen egyszerű (Prim-algoritmus): legyen $G(V, E)$ összefüggő, irányítatlan gráf, valamint jelölje A a keresett feszítőfa csúcsainak halmazát, míg B az élek halmazát. Első lépésként válasszunk tetszőleges csúcsot V-ből, majd töröljük V-ből, és kerüljön A-ba. Válasszuk ki a legkisebb élköltségű (v_j, v_i) élt úgy, hogy $v_i \in V$ és $v_j \in A$. A kiválasztott (v_j, v_i) élt tegyük át B-be, és v_j-t töröljük V-ből, és tegyük A-ba. Ha már G gráf minden csúcsa A-ban van, akkor megkaptunk egy olyan feszítőfát (tehát nem feltétlenül egyértelmű megoldáshoz jutunk), melynek éleit B tartalmazza. Ennek az eljárásnak igen nagy szerepe van többek között közüzemi hálózatok telepítésében.Rövid kitérő erejéig meg kell említenünk a témával kapcsolatosan a minimális feszítőfa problémát, mely egy összefüggő, irányítatlan gráfban a legkisebb összélköltségű feszítőfát keresi. (Feszítőfa alatt azt a fát

értjük, amely a gráf összes csúcsát tartalmazza, élei a gráf eredeti élei, és minden csúcsból, minden csúcsba pontosan egy út vezet). A problémára már 1926-ban adott egy megoldást Boruvka [9], melynek egy egyszerűsített változatát írta meg Jarník 1929-es levelében Boruvkának, majd 1930-ban cseh nyelven cikk formájában is megjelent [53]. Ám ez feledésbe merült, és tőle függetlenül Prim 1957-ben [71], valamint Dijkstra 1959-ben ismét megalkották az eljárást [29], ezzel sikerült javítaniuk Kruskal 1956-ban megjelent megoldásának számításigényén [36], melyet Boruvka nyomán írt. Az eljárás igen egyszerű (Prim-algoritmus): legyen $G(V, E)$ összefüggő, irányítatlan gráf, valamint jelölje A a keresett feszítőfa csúcsainak halmazát, míg B az élek halmazát. Első lépésként válasszunk tetszőleges csúcsot V-ből, majd töröljük V-ből, és kerüljön A-ba. Válasszuk ki a legkisebb élköltségű (v_j, v_i) élt úgy, hogy $v_i \in V$ és $v_j \in A$. A kiválasztott (v_j, v_i) élt tegyük át B-be, és v_j-t töröljük V-ből, és tegyük A-ba. Ha már G gráf minden csúcsa A-ban van, akkor megkaptunk egy olyan feszítőfát (tehát nem feltétlenül egyértelmű megoldáshoz jutunk), melynek éleit B tartalmazza. Ennek az eljárásnak igen nagy szerepe van többek között közüzemi hálózatok telepítésében.

Visszatérve a legrövidebb út problémához, Dijkstra algoritmusa után sok heurisztikus megoldás szü1etett a teljesítmény javítására. (A heurisztika minden esetben egy függvény, mely rangsorolja a lehetséges megoldásokat az elérhető információk alapján, ezzel segítve a továbblépésnél a gyorsabb döntést). Talán a legismertebb útkereső algoritmus, az A* (A-star) is ekkor született 1968-ban a Stanford Research Institute-ban, mely az első legjobb keresési (best-first search) eljárást használja heurisztikaként minden iterációban [68], hogy a lehető leghamarabb megtalálja az optimális utat, lásd Hart et al. [51]. Egyéb heurisztikus megoldások, mint például a B* (B-star) [7] vagy a kétirányú keresés (bi-directional search) után [16], 1987-ben sikerült a számítási igény terén áttörést elérnie Fredmannak és Tarjannak [39], Fibonacci-halmokon (F-heaps) alapuló, új adatstruktúrájuknak köszönhetően. A hálózatok bonyolultságának növekedésével nehezen tudta az informatika fejlődése tartani a versenyt, így 2005-ben a 9. alkalommal megrendezett Dimacs Challange [98] nevű tudományos verseny a legrövidebb út témájában írta ki pályázatát, és mintaadatként rendelkezésre bocsátották az USA akkori teljes úthálózatának gráfját. A verseny igen sok új eredményt generált, közülük is kiemelkedő a Karlsruhe Institue of Technology csapata által publikált cikkek sora. A rövidség kedvéért csak egyet, Geisberger et al. [46] cikkét emelném ki, akiknek érdeme abban rejlik, hogy a korábbi eredményeket javítani tudták azzal, hogy a keresés során nem preferált elemeket előzetesen eltávolítják a gráfból. Eljárásuknak a rövidítési rangsor (contraction hierarchy) nevet adták. Gyorsabb megoldásuknak azonban igen komoly előkalkuláció az ára. Ennek kiváltására tett kísérletet Delling et al. [27] RAPTOR nevű algoritmusa, mely egyáltalán nem igényel előkalkulációt, és mivel nem Dijkstra-algoritmusán alapszik, minden útat maximum egyszer vesz figyelembe iterációnként. Az előkalkulációk elhagyásával az algoritmus alkalmassá vált online alkalmazásokban való felhasználásra, hogy Pareto-optimális utakat kalkuláljon tömegközlekedési hálózatok felhasználói számára. 2013-as cikkében Dibbelt et al. [28] közzétettek Connection Scan nevű algoritmusukat, mely bár nem sokkal gyorsabb, mint a RAPTOR, de lényegesen egyszerűbb, mindamellett képes kezelni komplex eseteket is, például a várható késéseket, és ezt figyelembe véve kalkulálja a felhasználók várható érkezési idejét. A témakörben a 90-es évek közepéig megalkotott algoritmusokat részletesen tárgyalja Cherkassky et al. [20], különös figyelmet fordítva azok számításigényére.

A legrövidebb út problémára adott megoldások történeti áttekintése után térjünk rá az útvonaltervező eljárások gyakorlati problémákon való alkalmazásaira.

3. Útvonaltervező eljárások

Az egyik első útvonaltervező alkamazás az utazóügynök probléma (Traveling Salesman Problem, röviden TSP), melyet először az 1930-as években Karl Menger formalizált, és adott rá megoldást [64]. Lényege, hogy az ügynöknek adott telephelyeket kell felkeresnie, és dönteni csak arról tud (az élköltségek ismeretében), milyen sorrendben teszi ezt, hogy a lehető legkisebb költséggel járja körbe a telephelyeket. Tehát minimális összköltségű Hamilton-kört keresünk a gráfon. Birkhoff [8] munkájának köszönhetően lehetővé vált a hozzárendelési feladatok megoldása lineáris programozási feladatként, melyet Dantzig, Fulkerson és Johnson alkalmazott elsőként a TSP megoldására [24]. 1954-es cikkükben olyan módszereket vezetnek be, mely ma kombinatorikus optimalizálás alapját képezik, mint például a metszősíkok módszere. Fontos megemlítenünk, hogy a kombinatorikai és gráfelméleti alapok megteremtéséből olyan magyar tehetségek vették ki részüket, mint Kőnig Dénes a páros gráfok ekvivalencia tételével [59], majd tanítványa, Gallai Tibor független- és lefogó halmazokról szóló tételével [40], és Egerváry Jenő, aki általánosította a Kőnig-tételt [31], majd később a szállítási feladat kapcsán is elért önálló eredményt [32]. A magyar gráfelméleti iskola jelentőségét az is jól mutatja, hogy Kuhn Magyar-módszernek nevezte el az Egerváry munkája nyomán megalkotott, ma is alapvető eljárását a hozzárendelési feladat kombinatorikai megoldására [61]. A témakör tudománytörténeti hátterét bővebben Schrijver dolgozta fel [77].

A későbbiekben is javarészt ipari és gazdasági motivációk vezérelték a kutatások fókuszát, így alakult önálló témakörré a szállítmányozás tervezését segítő jármű útvonaltervezési probléma (Vehicle Routing Problem, röviden VRP), mely egy teherszállító flotta járműveinek telephely központú körútjainak optimalizálását célozza idő- és kapacitáskorlátok mellett. A probléma első formalizálására Dantzig és Ramser 1959-es cikkében került sor [26]. Később ennek több változata alakult ki: jármű útvonaltervezési probléma időablakokkal (Vehicle Routing Problem with Time Windows, röviden VRPTW), a korlátozó feltételek kibővültek a meglátogatandó célállomások nyitvatartási idejével, vagy éppen a kapacitáskorlátos jármű útvonaltervezési probléma esetén a szállítóeszköz kapacitás korlátjával

(Capacitated Vehicle Routing Problem, röviden CVRP), de több példát láthatunk a feltételek könnyítésére is: a többutas jármű útvonaltervezési probléma esetén a teherautók akár több körutat is tehetnek (Vehicle Routing Problem with Multiple Trips, röviden VRPMT), vagy nem feltétlenül szükséges az út végén a telephelyre visszatérniük a nyílt jármű útvonaltervezési problémában (Open Vehicle Routing Problem, röviden OVRP). Mivel a probléma NP-nehéz, így az idők során megannyi közelítő módszer született, ezek egyik jellemző iránya a heurisztikus megoldások köre:

- Genetikus algoritmusok, melyek utánozzák a mikrobiológusok által megfigyelt DNS-lánc javításának mechanizmusát, és az első fázisban - jellemzően mohó algoritmus segítségével - elkészült utakat variálják cserék és eltolások sorozatával. Az algoritmus futási ideje erősen függ attól, milyen megállási értéket állítanak be az algoritmusban (vagyis hány olyan random próbát tehet az algoritmus egymás után, ami nem javította a célfüggvény értékét, mielőtt új helyen próbál javulást elérni az algoritmus), lásd Chang és Chen [17].
- A hangya kolóniák módszere (ant colony system) a hangyák „motivációs eljárását” igyekszik utánozni: tudvalevő, hogy a hangyák feromonok segítségével kommunikálnak egymással. Amennyiben egy hangyának hosszú utat kell megtenni az élelem forrásáig, úgy egyre gyengül a feromon jel, amit maga után hagy. Ha azonban sikerül rövid utat találnia, ez a jel erős marad, így mind többen járnak majd a megtalált rövid úton. Ezt a logikát alkalmazták Bullnheimer et al. [12] VRP feladat megoldására.
- A szimulált lehűlés (simulated annealing) egy sztochasztikus technika, mely minden lépésben dönt - megfelelő kritériumok mellett -, hogy egy másik állapotba lépjen-e át, vagy helyben maradjon. A kohászatból vett kifejezés arra utal, ahogyan a fémet ellenőrzött körülmények között felhevítik, majd visszahűtik, hogy a szerkezetét erősítsék, és a benne található zárványokból minél több eltűnjön. Ezzel az eljárással keres globális optimumot VRPTW feladatra Czech és Czarnas [23].
- A tabu keresés (tabu search) megoldások a memóriában tárolják azokat a megoldásokat, melyek korábbi iterációkban tesztelve lettek és valamilyen előre megállapított szabály miatt tiltó listára kerültek (egy időre). Az eljárást például Bräysy és Gendreau alkalmazta VRPTW megoldására [11].
- A 2-opt általában más algoritmusokkal kombinálva jelenik meg a megoldásokban. Lényege, hogy olyan út, mely keresztezi saját magát, úgy legyen átrendezve, hogy ne legyen benne kereszteződés. Az algoritmus leírását elsőként Croes adta 1958-ban a TSP megoldására [22].
- A 3-opt olyan lokális keresési (local search) algoritmus, mely a gráfon, vagy úton 3 szomszédos csúcsot töröl, majd ezeket minden lehetséges módon újra-

rendezve igyekszik az optimális utat, vagy utakat megtalálni. Az algoritmust elsőként Lin formalizálta 1965-ben [57].

- A Lin-Kernighan-algoritmus a 2-opt és 3-opt eljárások általánosítása, melyben mindkét algoritmust adaptívan alkalmazzuk az útvonalakon. A Lin és Kernighan [58] által 1973-ban alkotott algoritmus az egyik leghatékonyabb eljárás a TSP megoldására.

Mindemellett egzakt algoritmusok is születtek:

- Korlátozás és szétválasztás (branch and bound) egy kombinatorikus optimalizációs eljárás, szétválasztás szakaszában a keresési halmazt diszkrét halmazokra bontja bizonyos szabályok alapján, majd a korlátozás szakaszban az egyes halmazokat „ritkítja”, ezzel gyorsítva fel a keresést a brute-force megoldásokhoz képest, lásd Bektas et al. [6]
- A vágás és szétválasztás (branch and cut) eljárás egészértékű lineáris programozási (Integer Linear Programming, röviden ILP) feladatok megoldására szolgál, melynek keretében először a korlátozás és szétválasztás algoritmust használjuk az LP feltételeinek könnyítésére, majd metszősíkok módszerével szűkítjük azokat, hogy az optimumhoz közelebb jussunk. Jó példa ennek alkalmazására VRP feladat megoldásában Pessoa et al. [69].
- Az egzakt algoritmusok számításigénye gyakran csökkenthető olyan eljárásokkal, melyek egyszerű megfontolások alapján az irreleváns csúcsokat, vagy csúcskombinációkat eleve törlik. Erre jó példa Lu et al. [63] Trip-Mine algoritmusa, ahol a csúcsok költség-profit alapú rendezésével, valamint már időben el nem érhető csúcsok törlésével lerövidítik a vizsgálandó esetek számát. Így a vizsgált „brute force” algoritmus (mely 12 csúcs kalkulálása esetén már majdnem 1 órás futási időt produkál, hiszen 12! esetet kell leszámlálnia) helyett javasolt eljárás néhány ezred másodpercre csökkenti annak futásidejét.
- Mivel a VRPTV formalizálható egyenletrendszerként, így a probléma LP feladatként való megoldása is lehetséges, lásd Rousseau et al. [75].

Az utazóügynök problémából kifejlődő modellek másik ága a tájfutó problémája (*Orienteering Problem*, röviden OP), vagy más néven a szelektív utazóügynök probléma (*Selective Traveling Salesman Problem*, röviden STSP), ahol az egyes ügyfelekhez már profitot rendelnek, és az ügynököt szorító időkorláton belül a legnagyobb összprofitot kell begyűjtenie az útja során az ügyfelek meglátogatásával. Az elnevezés 1996-ban Chao et al. [18] cikkében szerepel, de már 1984-ben megjelent Tsiligirides-nél [88], ahol a TSP-ben az ügynöknek nincs elég ideje, hogy az összes várost meglátogassa egyedül. Cikkében olyan sztochasztikus algoritmust alkalmaz az optimális útvonal közelítő megoldására, mely minden iterációban Monte-Carlo-módszerrel keresi a következő csúcsot, a távolság és a begyűjthető profit függvényében. A problémát már formalizálta Kataoka és

Morito 1988-ban [55], ám ők még maximális gyűjtési probléma (*Maximum Collection Problem*) néven hivatkoztak rá. A témáról bővebben Feillet et al. összefoglaló cikkében olvashatunk [35]. Az OP megfogalmazását az A.1. függelék alatt találjuk. Már a kezdetektől ismert volt ennek a technikának a természetjárásban és általában a turizmusban való alkalmazhatósága, hiszen az OP elnevezés is a tájfutásból ered, ahol a versenyzőknek egy térkép és egy iránytű segítségével kell felkeresni az előre kijelölt pontokat a lehető legrövidebb időn belül. Innen datálható a tudományág sport és turizmus területén történő hasznosítása, és terjedt ki nem csak a természetjárásra, de a városnézésre is. Ennek jó példája Wang et al. [95], ahol a legérdekesebb látványosságokat látogatja végig a turista a szállodábol indulva, és a nap végén oda visszaérkezve. Golden, Levy és Vohra megmutatták, hogy az OP NP-nehéz [49], így az erre adott egzakt megoldások csak viszonylag kis számú csúcs esetén lehetséges. Ramesh et al. [73] korlátozás és szétválasztás algoritmust használ, mellyel egzakt megoldást ad akár 150 csúcsot tartalmazó gráfra is, míg Fischetti et al. [36] cikkükben brach-and-bound eljárással akár 500 csúcsra is egzakt megoldást tudnak adni. Ramesh és Brown [72] 4 fázisból álló heurisztikus megoldást adnak az OP-re, melyben a 2-opt és 3-opt eljárásokat alkalmazzák. Ennél jobb eredményeket ad Chao et al. [18] 5 lépésből álló megoldása, mely mohó algoritmust, sztochasztikus eljárást és opt-2 algoritmust ötvözve építi fel az útvonalat. A fenti heurisztikus megoldások komoly hátránya, hogy könnyen be tudnak ragadni egy lokális optimumba, melyet Gendreau et al. [48] tabu search megoldása hatékonyan hidal át. Mivel az eredmények turisztikában történő felhasználása igen nagy figyelmet kap, így cikkek sora foglalkozik azok térinformatikai beágyazásával is (mobil applikációk formájában), erre jó példát találunk az OP esetére Souffriau et al. 2008-as cikkében [80]. A tájfutó problémája időablakkal (*Orienteering Problem with Time Windows*, röviden OPTW) az OP általánosítása, ahol a csúcsokhoz nyitvatartási időket rendelünk. Az OPTW leírását az A.2. függelék alatt adjuk meg. Elsőként Kantor és Rosenwein [54] adtak rá megoldást 1992-ben. Első lépésben úgy illesztenek be az útvonalba új csúcsokat, hogy ne ütközzön időkorlátba, és az egységnyi időköltségre eső fajlagos profit a lehető legnagyobb legyen. Ezután mélységi keresési algoritmussal állítanak elő útszakaszokat, majd fűzik őket össze, elhagyva a nem megvalósítható elemeket. Mivel az időablakok miatt az OP-nél hatékonyan alkalmazható opt-2 és opt-3 algoritmusok OPTW esetén nem használhatóak, így annak egzakt megoldására más eljárásra van szükség. Az azonban igaz, hogy az OPTW megoldására használt eljárás alkalmazható az OP megoldására. Ezt megmutatják Tricoire et al. [87] 2010-es cikkükben. Righini és Salani 2009-es cikkében cite72 kétirányú dinamikus programozási megoldást javasol: a kezdő- és végcsúcstól egyszerre kezdik el az út felépítését, végig ellenőrizve, hogy megvalósítható-e az egyes lépésekben javasolt megoldás, ha a két szakaszt összekapcsolnánk.

Az OP egy természetes kiterjesztése a tájfutó csapat probléma (*Team Orienteering Problem*, röviden TOP), ahol a turista „feladata”, hogy P nap alatt a rendelkezésére álló időben a lehető legtöbb (számára érdekes) látványosságot meglátogasson, és minden nap végén visszatérjen a szállodájába, (ez igen hasonlít a VRPTW-ben megfogalmazott feladathoz). Ezt először Butt és Cavalier formalizálta 1994-ben [13], ahol egy toborzási feladat megoldására alkalmazták. A TOP megfogalmazását az A.3. függelék alatt találjuk. Az egzakt megoldások közül igen hatékonyan működnek az oszlopgeneráló algoritmuson [4] alapuló eljárások. Ekkor LP feladatként oldjuk meg a feladatot, de redukáljuk a dimenziók számát a gyorsabb futási idő érdekében, melyre jó példa Butt és Ryan 1999-es cikke [14], ahol akár 100 csúcsra is egzakt megoldást kaphatunk viszonylag rövid idő alatt. Később Boussier et al. [10] alkalmazta az oszlopgeneráló algoritmust, de már kombinálva a korlátozás és szétválasztás eljárással, hogy javítsanak az algoritmus teljesítményén. A heurisztikus megoldások közül a legkorábbi a már az OP kapcsán ismertetett Chao et al. [18] cikkében szereplő 5 lépcsős eljárás kis átalakítással: itt az első P legjobb utat listázzuk ki eredményül [19]. Tang és Miller-Hooks [84], valamint Archetti et al. [3] is tabu keresés eljárást alkalmaz a TOP megoldására, míg Ke et al. [56] hangya kolóniák módszerét javasolja cikkében. Az első lépésben 4 eljárást is teszteltek, amivel egy megvalósítható eljáráshoz lehet jutni. Közülük az utakat szekvenciálisan felépítő algoritmus bizonyult a leghatékonyabbnak. Az egyes iterációkban elkészült megoldást 2-opt algoritmussal javítják, majd kiegészítik annyi csúccsal, amennyi az időkorlátba belefér. Vansteenwegen et al. két heurisztikus eljárást is kifejlesztett. Mind az irányított lokális keresés (*Guided Local Search*, röviden GLS) [90], mind a ferde változó szomszéd keresés (*Skewed Variable Neighborhood Search*, röviden SVNS) [92] eljárások ugyanazokon a lépéseken alapulnak: egy kezdeti eljárásból kiindulva „gyengébb” útszakaszokat törlünk, illetve kisebb útszakaszokat illesztünk össze, majd az így kapott út összprofitját igyekszik javítani cserékkel, illetve a menetidőket csökkenteni, és új pontokat beilleszteni a megtakarított idő terhére. Az SVNS más sorrendben variálja ezeket a lépéseket, és így jóval megelőzi a GLS-t. A TOP időablakokkal általánosított változata a tájfutó csapat problémája időablakkal (*Team Orienteering Problem with Time Windows*, röviden TOPTW), melynek leírását az A.4. függelékben adjuk meg. A TOPTW-re adott megoldások közül Vansteenwegen et al. [91] iterált lokális keresés algoritmusa (*Iterated Local Search*, röviden ILS) messze a leggyorsabb, bár akadnak eljárások, melyek átlagosan kicsivel jobb megoldást adnak. Ilyen például Gambardella et al. [41] hangya kolóniák módszerén alapuló eljárása. Tricoire et al. [87] a TOPTW egy általánosítására, a többperiódusos, több időablakos tájfutó problémájára (*Multi-Period Orienteering problem with Multiple Time Windows*, röviden MPOPMTW) ad heurisztikus megoldást változó szomszéd kereső eljárással (*Variable Neighborhood Search*, röviden VNS) [50] eljárással, míg az útvonal megvalósíthatóságának ellenőrzésére egzakt algoritmust javasolnak. Ez esetben az egyes telephelyeknek napok között változó lehet a nyitvatartási ideje. Kísérleteik alapján 100 csúcs és 2 megtervezendő út esetén nagyjából 1 perc alatt jut megoldásra, míg Vansteenwegenék ILS algoritmusával ez 1 másodperc.

4. Az útvonaltervező eljárások néhány kiterjesztése

A fent ismertetett modelleknek több lehetséges általánosítása létezik, melyek közül a teljesség igénye nélkül néhányat megemlítünk az alábbiakban:

- Az időfüggő tájfutó problémája (*Time-dependent OP*, röviden TDOP) lényege, hogy az egyes élköltségek időben változnak. Jól írja le ez a modell azt a gyakorlati problémát, hogy napszakonként eltérő a városi közlekedés minősége: változik a forgalom és a tömegközlekedési eszközök járatsűrűsége is. Ez talán akkor érint bennünket legkevésbé, ha csak gyalogosan közlekedünk a városban, bár a lámpák beállításai még így is időben változó módon befolyásolja menetidőnket, lásd Fomin és Lingas [37]. Verbeeck et al. [94] hangya kolóniák módszerét kombinálta lokális kereső eljárásokkal a TDOP megoldására. Korábban Abbaspour és Samadzadegan [1] adnak közelítő megoldást a TDOPTW-re genetikus algoritmus segítségével. Az ILS jó kompromisszumot nyújt gyorsaság és pontosság között, de minden csúcsot külön kezel. Az időfüggő, időablakos tájfutó csapat problémája (*Time-dependent Team Orienteering Problem with Time Windows*, röviden TDTOPTW) megoldása a hagyományos ILS módszerrel már nem lenne hatékony, így García et al. [43] előkalkulációs eljárással visszavezeti TOPTW feladatra, majd ILS algoritmussal oldja meg azt. Egy másik módszert is kidolgoztak, mely nem él az időbeni függés eliminálásával, ám helyette a tömegközlekedés menetrendjére tesznek periodicitási feltevéseket (mely koránt sem realisztikus). Gavalas et al. [44] javasolja az egymáshoz közel eső pontok együtt kezelését a probléma egyszerűsítése érdekében, melyhez k-közép klaszterező (k-means clustering) eljárást alkalmaznak. Athéni helyszíneket és tömegközlekedést modellező kutatásukban klasztereken alapuló heurisztikus eljárásukat tovább fejlesztve 3 algoritmust is adnak a TDTOPTW közelítésére, melyek az időablakok mellett kezelni tudják az időben változó utiköltségeket és a tömegközlekedési menetrendet is [45]. Az eljárásaik hátránya, hogy nem veszik figyelembe az újabb csúcsok útvonalba történő beillesztésénél a következő helyszín várakozási idejében okozott változást, mikor a beillesztésről döntenek.

- Az általánosított tájfutó problémája (*Generalized Orienteering Problem*, röviden GOP) abban különbözik az OP-től, hogy célfüggvénye nem pusztán a csúcsokban begyűjthető profitok összessége, hanem általánosabb, nemlineáris összefüggés a pontok között. Lehet például az egyes helyszínek változatosságát extra profittal értékelni (például a negyedik múzeum meglátogatása helyett egy park felkeresése esetén), vagy bizonyos kiegészítő helyszínek megtekintése, például Glasgow-ban Mackintosh múzeummá alakított házának meglátogatása után érdemes felkeresni az általa tervezett Willow Tearooms enteriőrjét. Schilde et al. [76] cikkében a turisták különleges igényeit próbálja leírni nemlineáris célfüggvényekkel. Az Aurigo nevű alkalmazás |[97] útvonaltervező algoritmusa igen egyszerű, hiszen csak az épp adott tartózkodási hely egy r

sugarú környezetében keresi a következő, legnagyobb profitú pontot, de a profitok adaptív módon, dinamikusan kerülnek meghatározásra a felhasználó ízlése, valamint a már meglátogatott pontok függvényében.

- Cikkek sora foglalkozik olyan modellekkel, ahol az egyes élekhez profitok vannak rendelve. Amennyiben a csúcsokhoz nincs, csak az élekhez, azt a szakirodalomban él útvonaltervező probléma (*Arc Routing Problem*, röviden ARP) néven találjuk. A feladat, hogy két adott pont között a lehető legtöbb profitot begyűjtve haladjunk át éleken, melyeknek költség vonzata is van. Souffriau et al. [81] például az észak-flandriai úthálózaton tesztelte biciklis útvonaltervező mohó véletlenszerű adaptív keresési eljárásukat (*Greedy Randomized Adaptive Search Procedure*, röviden GRASP) eljárását, mely előbb mohó algoritmussal jut egy kezdeti megoldáshoz, majd azt javítja a következő lépésben lokális keresési eljárással. Muyldermans et at. [67] az OP-t kiegészítve élekhez rendelt profitokkal formalizálta az általuk általános útvonaltervezési problémának (*General Routing Problem*, röviden GRP) nevezett feladatot, majd adott rá egzakt megoldást opt-2 és opt-3 algoritmusok felhasználásával. A feladat gyakorlati jelentősége a turisztikai célú útvonaltervezésben az lehet, hogy ezáltal a szebb, látványosabb útvonalakat, mint például a sugárutak vagy folyópartok, előnyben részesíthetjük.
- Ha az OP-ben egyes csúcsokat kötelezővé teszünk, az az általános tájfutó problémában (*Generalized Orienteering Problem*, röviden GOP) egy szélsőséges alesetnek tekinthető (végtelenül nagy profitokat rendelve bizonyos csúcsokhoz). Gendreau et al. [47] ilyen eljárással biztosítja, hogy a legfontosabb látnivalók minden egyedileg tervezett túraútban benne legyenek.
- Amennyiben az egyes csúcsoknál begyűjthető profitok értéke előre nem ismert, csupán azok eloszlásáról van tudomásunk, az OP-ben megismert feladatunk annyiban módosul, hogy az összprofitunk várható értékét kell maximalizálnunk, melyet sztochasztikus profitú tájfutó probléma (*Orienteering Problem with Stochastic Profits*, röviden OPSP) néven találunk a szakirodalomban. Például Ilhan et al. [52] genetikus algoritmust adott az optimum közelítésére, valamint egy egzakt megodást is, melyben a sztochasztikus célfüggvényt vele ekvivalens, determinisztikus célfüggvényre cserélik, majd súlyozott összeg eljárással (weigthed sum method) [86] oldják meg a feladatot.
- A csúcsoknál gyűjthető profitok értéke lehet időben változó, de ismert érték. Ez főleg szállítási feladoknál fordul elő, ahol a késedelmes kiszállítás büntetéssel járhat. Erre adott eljárást Erkut és Zhang [34], ahol a szállítási feladatot időfüggő díjazású maximális gyűjtési probléma (*Maximum Collection Problem with Time Dependent Rewards*, röviden MCPTDR) modellel írta le, és a profitok időben lineárisan csökkentek. Ezt egészértékű programozási feladatként kezelték, melyre korlátozás és szétválasztás algoritmussal és egy mohó

algoritmussal adtak közelítő megoldást. Ennek több útra felírt változatára (*Multiple Tour Maximum Collection Problem with Time-Dependent rewards*, röviden MTMCPTD) ad megoldást Tang et al. [85], akik hibaelhárító szerelőcsoportok kiszállásait optimalizálására tabu keresés algoritmust adtak közelítő megoldásként. A turizmusban olyan gyakorlati esetekben fordulhat elő, mikor egy kiállítás valamely részlege csak szűkebb látogatási időben érhető el, és annak zárva tartása esetén a csúcsnál gyűjthető profit értéke kisebb, vagy mint Erdogan és Laporte cikkében [33], ahol az adott ponton töltött időtől függ a beszedhető profit.

- A TOPTW egy másik általánosítása a szelektív jármű útvonaltervezési probléma időablakokkal (*Selective Vehicle Routing Problem with Time Windows*, röviden SVRPTW), ahol két új korlátot vezethetünk be: a járművek nem csak időkorlátokkal bírnak, de távolságkorláttal is, valamint a rakterükből adódó kapacitáskorláttal. Ezt tetszőlegesen értelmezhetjük turistákra is, akik egy bizonyos távolság megtétele után elfáradnak, valamint anyagi lehetőségük is véges, így nem tudnak naponta egy adott összegnél többet elkölteni a nevezetességeknél megváltandó belépőjegyekre. Boussier et al. [10] korábban említett egzakt algoritmusa erre a problémára is megoldást ad akár 100 csúcs és 10 megtervezendő út esetére is.

- Ennek egy speciális változata a kapacitáskorlátos tájfutó csapat probléma (*Capacitated Team Orienteering Problem*, röviden CTOP), ahol csak egy extra kapacitáskorláttal (pénzügyi korlát) egészítjük ki a TOP modelljét, lásd Archetti et al. [2].

- A turizmusban előforduló gyakorlati problémából fakad a szálloda választó tájfutó probléma (*Orienteering Problem with Hotel Selection*, röviden OPHS), ami a TOP feladat kibővítve azzal, hogy egy adott halmazból szállást kell választani az utakhoz (ahol azok kezdődnek és végződnek), lásd Divsalar et al. [30]. Castro et al. [15] a TSP-t egészíti ki szállodaválasztással (TSPHS), melyre ILS és egy speciális genetikus algoritmus kombinációjából álló heurisztikus megoldást adnak cikkükben.

- Külön említést érdemel még az útvonaltervező feladatok egy speciális családja, mely a turisták gyakorlati útvonaltervező feladatait kívánja megoldani, és gyakran köthető mobil alkalmazásokhoz, és ebből adódóan kis számításigényű eljárást kíván. Elnevezése, a turistaút tervezési probléma (*Tourist Trip Design Problem*, röviden TTDP), Vansteenwegen és Van Oudheusden 2007-es cikkéből származik [89]. A TTDP legegyszerűbb modellje az OP, és gyakorlati jelentőséget tulajdoníthatunk annak minden kiterjesztésének. A TTDP megoldások részletes áttekintését olvashatjuk Gavalas et al. [45] összefoglaló cikkében. A mobil eszközökre készült alkalmazások jó példája Sylejmani és Dika cikke [83], ahol Bécs turisztikai látványosságain tesztelték tabu search alapú heurisztikus algoritmusukat. García et al. [42] a TDTOPTW megoldására tesznek javaslatot heurisztikus algoritmusukkal, mely személyre szabott profitokkal látja el az egyes csúcsokat a felhasználó preferenciáinak megfelelően. A TDTOPTW mobil alkalmazásokra tervezett megoldások közül Souffriau et al. [82] ILS algoritmussal adott közelítése az egyik leghatékonyabb.

- Az útvonaltervező eljárások egy máshova kevéssé beilleszthető példája De Choudhury et al. [21] cikke, akik „közösségi kenyérmorzsáknak" (social breadcrumbs) nevezett információ alapján építenek túraútvonalakat. Az interneten (Facebook, Flickr stb.) megosztott fotók és egyéb bejegyzések gyűjtése és szisztematikus válogatása alapján, összeegyeztetve a felhasználó előre kinyilvánított preferenciáival. Mivel a fotókhoz időbélyegek (timestamp) is tartoznak, így Popescu és Grefenstette [70] korábbi munkája alapján már lehetőség nyílt az egyes helyszínek látogatási idejének, illetve a köztük megtett út menetidejének becslésére is. Hasonlóan közösségi adatokon alapszik Letchner et al. [36] munkája, akik helyi lakosok autós GPS adatai alapján jobb útvonalat tudtak javasolni az átutazóknak, mint amit bármilyen útvonaltervező adott, mert ők egy eddig fel nem használt információt építettek a tervezésbe: a tapasztalatot. A dolgozatban bemutatott útvonaltervezési problémák közötti kapcsolatot a B függelékben szemléltetjük

Az útvonaltervező algoritmusokról bővebb összefoglalót Vansteenwegen et al. cikkében olvashatunk [93], ahol külön kitérnek az egyes eljárások számítási igényére is.

5. Befejezés

A fentiekben ismertetett problémákon és azokra adott megoldások bonyolultságán láthatjuk, miként váltak az idők során egyre inkább életszerűvé és pontosabbá a modellek. Az eddigi problémák kiterjesztésének, valamint az újabb útvonaltervező eljárásoknak csak a fantázia és a rendelkezésre álló eszközök számítási teljesítménye szabhat határt. Nem kétséges, hogy még a mi életünkben több nagyságrenddel nagyobb bonyolultságú feladatok megoldásának lehetünk szemtanúi.

Hivatkozások

[1] R. A. Abbaspour, F. Samadzadegan: *Time-dependent personal tour planning and scheduling in metropolises*, Expert Systems and Applications, Vol. **38** (2011), 12439–12452. doi:10.1016/j.eswa.2011.04.025.

[2] C. Archetti, D. Feillet, A. Hertz, M. Speranza: *The capacitated team orienteering*

and profitable tour problems, Journal of the Operational Research Society, Vol. **60** (2009), 831–842. doi:10.1057/palgrave.jors.2602603

[3] C. Archetti, A. Hertz, M. Speranza: *Metaheuristics for the team orienteering problem*, Journal of Heuristics, Vol. **13** (2007), 49–76. doi: 10.1007/s10732-006-9004-0

[4] C. Barnhart, E. L. Johnson, G. L. Nemhauser, M. W. P. Savelsbergh, P. H. Vance: *Branch-and-Price: Column Generation for Solving Huge Integer Programs*, Operations Research, Vol. **46**, No. **3** (1998), 316–329.

[5] R. Bellman: *On a routing problem*, Quarterly of Applied Mathematics, Vol. **16** (1958), 87–90.

[6] T. Bektaş, G. Erdoğan, S. Røpke: *Formulations and Branch-and-Cut Algorithms for the Generalized Vehicle Routing Problem*, Journal of Transportation Science, Vol. **45**, No. **3** (2011), 299–316. doi: 10.1287/trsc.1100.0352

[7] H. Berliner: *The B* Tree Search Algorithm. A Best-First Proof Procedure*, Artificial Intelligence, Vol. **12**, No. **1** (1979), 3–40. doi:10.1016/0004-3702(79)90003-1

[8] [2] G. Birkhoff: *Tres observaciones sobre el algebra lineal*, Revista Facultad de Ciencias Exactas, Puras y Aplicadas Universidad Nacional de Tucuman, Serie A (Matematicas y Fisica Teorica), Vol. **5** (1946), 147–151.

[9] O. Boruvka: *O jistém problému minimálním [On a minimal problem]*, Práce Moravské Príirodovedecké Spolecnosti, Brno [Acta Societatis Scientiarum Naturalium Moravicae], Vol. **3** (1926), 37–58. (cseh nyelven, német előszóval)

[10] S. Boussier, D. Feillet, M. Gendreau: *An exact algorithm for the team orienteering problem*, 4OR, Vol. **5** (2007), 211–230. doi:10.1007/s10288-006-0009-1

[11] O. Bräysy, M. Gendreau: *Tabu Search Heuristics for the Vehicle Routing Problem with Time Windows*, Sociedad de Estadística e Investigación Operativa TOP, Vol. **10**, No. **2** (2003), 211–237. doi: 10.1007/BF02579017

[12] B. Bullnheimer, R.F. Hartl, C. Strauss: *Applying the Ant System to the Vehicle Routing Problem*, S. Voss, I.H. Osman, C. Roucairol (eds.): Meta-Heuristics: Advances and Trends in Local Search Paradigms for Optimization, Kluwer Academic Publishers Norwell (1999), 285–296. doi: 10.1007/978-1-4615-5775-3_20

[13] S. E. Butt, T. M. Cavalier: *A heuristic for the multiple tour maximum collection problem*, Computers and Operations research, Vol. **21** (1994), 101–111. *A heuristic for the multiple tour maximum collection problem*

[14] S. Butt, D. Ryan: *An optimal solution procedure for the multiple tour maximum collection problem using column generation*, Computers and Operations Research, Vol. **26** (1999), 427–441. doi:10.1016/S0305-0548(98)00071-9

[15] M. Castro, K. Sörensen, P. Vansteenwegen, P. Goos: *A fast metaheuristic for the travelling salesperson problem with hotel selection*, 4OR quarterly journal of the Belgian, French and Italian Operations Research Societies, Vol. **13**, No. **1** (2015), 15–34. DOI: 10.1007/s10288-014-0264-5

[16] D. de Champeaux, L. Sint: *An improved bidirectional heuristic search algorithm*, Journal of the ACM, Vol. **24**, No. **2** (1977), 177–191. doi:10.1145/322003.322004

[17] Y. Chang, L. Chen: *Solving the Vehicle Routing Problem with Time Windows via a Genetic Algorithm*, Discrete and Continuous Dynamical Systems Supplement (2007), 240–249. doi: 10.1016/j.eswa.2008.09.001

[18] I. Chao, B. Golden, E. Wasil: *Theory and methodology – a fast and effective heuristic for the orienteering problem*, European Journal of Operational Research, Vol. **88** (1996), 475–489. doi:10.1016/0377-2217(95)00035-6

[19] I. Chao, B. Golden, E. Wasil: *Theory and methodology – the team orienteering problem*, European Journal of Operational Research, Vol. **88** (1996), 464–474. doi:10.1016/0377-2217(94)00289-4

[20] B. V. Cherkassky, A. V. Goldberg, T. Radzik: *Shortest paths algorithms: theory and experimental evaluation*, Mathematical Programming, Vol. **73**, No. **2** (1996), 129–174. doi: 10.1007/BF02592101

[21] M. De Choudhury, M. Feldman, S. Amer-Yahia, N. Golbandi, R. Lempel, C. Yu: *Automatic construction of travel itineraries using social breadcrumbs*, in Proceedings of the 21st ACM conference on Hypertext and Hypermedia (2010), 35–44. doi: 10.1145/1810617.1810626

[22] G. A. Croes: *A method for solving traveling salesman problems*, Operations Research, Vol. **6** (1958), 791–812. doi: 10.1287/opre.6.6.791

[23] Z. J. Czech, P. Czarnas: *Parallel simulated annealing for the vehicle routing problem with time windows*, in Proceesings of the 5th International Conference (2003), 233–240. doi: 10.1109/EMPDP.2002.994313

[24] G. Dantzig, R. Fulkerson, S. Johnson: *Solution of a Large Scale Traveling Salesman Problem*, Journal of the Operations Research Society of America, Vol. **2** (1954), 393–410. doi: 10.1287/opre.2.4.393

[25] G. B. Dantzig: *On the Shortest Route through a Network*, The RAND Corporation, Santa Monica, California, paper P-1345. Published in Management Science, Vol. **6** (1958), 1960, 187–190.

[26] G. B. Dantzig, J. H. Ramser: *The Truck Dispatching Problem*, Management Science, Vol. **6** (1959), 80–91. doi: 10.1287/mnsc.6.1.80

[27] D. Delling, T. Pajor, R. F. Werneck: *Round-Based Public Transit Routing*, in Proceedings of the Sixth International Symposium on Combinatorial Search (2012), 130–140. doi:10.1287/trsc.2014.0534

[28] J. Dibbelt, T. Pajor, B. Strasser, D. Wagner: *Intriguingly Simple and Fast Transit Routing*, in Proceedings of the 12th International Symposium on Experimental Algorithms (2013), 43–54. doi: 10.1007/978-3-642-38527-8_6

[29] E. W. Dijkstra: *A note on two problems in connexion with graphs*, Numerische Mathematik, Vol. **1** (1959), 269–271. doi:10.1007/BF01386390

[30] A. Divsalar, P. Vansteenwegen, D. Cattrysse: *A variable neighborhood search method for the orienteering problem with hotel selection*, International Journal of Production Economics, Vol. **145**, No. **1** (2013), 150–160. doi:10.1016/j.ijpe.2013.01.010

[31] J. Egerváry: *Mátrixok kombinatorius tulajdonságairól*, Matematikai és Fizikai Lapok, Vol. **38** (1931), 16–28.

[32] E. Egerváry: *Bemerkungen zum Transportproblem*, MTW Mitteilungen, Vol. **5** (1958), 278–284.

[33] G. Erdogan, G. Laporte: *The orienteering problem with variable profits*, Networks, Vol. **61**, No. **2** (2013), 104–116. DOI: 10.1002/net.21496

[34] E. Erkut, J. Zhang: *The maximum collection problem with time-dependent rewards*, Naval Research Logistics, Vol. **43**, No. **5** (1996), 749–763, DOI: 10.1002/(SICI)1520-6750

[35] D. Feillet, P. Dejax, M. Gandreau: *Traveling Salesman Problems with Profits*, Journal of Transportation Science, Vol. **39**, No. **2** (2004), 188–205. DOI: 10.1287/trsc.1030.0079

[36] M. Fischetti, J. Salazar, P. Toth: *Solving the orienteering problem through branch-and-cut*, INFORMS Journal on Computing, Vol. **10** (1998), 133–148. doi: 10.1007/978-3-319-18161-5_17

[37] F. V. Fomin – A. Lingas: *Approximation Algorithms for Time-Dependent Orienteering*, Lecture Notes in Computer Science, Vol. **2138** (2001), 508–515., doi: 10.1007/3-540-44669-9_57

[38] L. R. Ford, Jr: *Network Flow Theory, The RAND Corporation*, Santa Monica, California, paper P-923, (1956).

[39] M. L. Fredman, R. E. Tarjan: *Fibonacci Heaps and Their Uses in Improved Network Optimization Algorithms*, Journal of the Association for Computing Machinery, Vol. **34**, No.**3** (1987), 596–615. doi:10.1109/SFCS.1984.715934

[40] T. Gallai: *Maximum-minimum Satze uber Graphen*, Acta Mathematica Academiae Scientiarum Hungaricae, Vol. **9** (1958), 395–434.

[41] L. M. Gambardella, R. Montemanni, D. Weyland: *Coupling ant colony systems with strong local searches*, European Journal of Operational Research, Vol. **220**, No.**3** (2012), 831–843. DOI: 10.1016/j.ejor.2012.02.038

[42] A. Garcia, O. Arbelaitz, M. T. Linaza, P. Vansteenwegen, W. Souffriau: *Personalized tourist route generation*, in Proceedings of the 10th International Conference on Current Trends in Web Engineering (2010), 486–497. doi: 10.1007/978-3-642-16985-4_47

[43] A. Garcia, P. Vansteenwegen, O. Arbelaitz, W. Souffriau, M. T. Linaza: *Integrating public transportation in personalised electronic tourist guides*, Computers & Operations Research, Vol. **40**, No. **3** (2013), 758–774. doi:10.1016/j.cor.2011.03.020

[44] D. Gavalas, C. Konstantopoulos, K. Mastakas, G. Pantziou, Y. Tasoulas: *Cluster-based heuristics for the team orienteering problem with time windows*, in Proceedings of 12th International Symposium on Experimental Algorithms (2013), 390–401. doi: 10.1007/978-3-642-38527-8_34

[45] D. Gavalas, C. Konstantopoulos, K. Mastakas, G. Pantziou, N. Vathis: *Heuristics for the Time Dependent Team Orienteering Problem: Application to Tourist Route Planning*, Computers & Operations Research, Vol. **62** (2015), 36–50. doi:10.1016/j.cor.2015.03.016

[46] R. Geisberger, P. Sanders, D. Schultes, D. Delling: *Contraction Hierarchies: Faster and Simpler Hierarchical Routing in Road Networks*, in Proceedings of the 7th international conference on Experimental Algorithms, WEA'08 (2008), 319–333 doi: 10.1007/978-3-540-68552-4_24

[47] M. Gendreau, G. Laporte, F. Semet: *A branch-and-cut algorithm for the undirected Selective Travelling Salesman Problem*, Networks, Vol. **32** (1998), 263–273. DOI: 10.1002/(SICI)1097-0037(199812)32:4<263::AID-NET3>3.0.CO;2-Q

[48] M. Gendreau, G. Laporte, F. Semet: *A tabu search heuristic for the undirected selective travelling salesman problem*, European Journal of Operational Research, Vol. **106** (1998), 539–545. doi:10.1016/S0377-2217(97)00289-0

[49] B. Golden, L. Levy, R. Vohra: *The Team Orienteering Problem*, Naval Research Logistics Quarterly, Vol. **34** (1984), 307–318. DOI: 10.1002/1520-6750(198706)34:3<307::AID-NAV3220340302>3.0.CO;2-D

[50] P. Hansen, N. Mladenovic, J. A. M. Perez: *Variable neighbourhood search: methods and applications*, Annals of Operations Research, Vol. **175** (2010), 367–407. doi: 10.1007/s10288-008-0089-1

[51] s̃c P. E. Hart, N. J. Nilsson, B. Raphael: *A Formal Basis for the Heuristic Determination of Minimum Cost Paths*, Transactions on Systems Science and Cybernetics, Vol. **4**, No. **2** (1968), 100–107. doi:10.1109/TSSC.1968.300136

[52] T. Ilhan, S. Iravani, M. Daskin: *The orienteering problem with stochastic profits*, IIE Transactions, Vol. **40** (2008), 406–421. DOI:10.1080/07408170701592481

[53] V. Jarník: *O jistém problému minimálním (Egy bizonyos minimális problémáról)*, Práce Moravské Přírodovědecké Společnosti, Vol. **6** (1930), 57–63. (cseh nyelven)

[54] M. Kantor, M. Rosenwein: *The orienteering problem with time windows*, The Journal of the Operational Research Society, Vol. **43**, No. **6** (1992), 629–635. DOI: 10.2307/2583018

[55] S. Kataoka, S. Morito: *An algorithm for the single constraint maximum collection problem*, Journal of the Operations Research Society of Japan, Vol. **31**, No. **4** (1988), 515–530.

[56] L. Ke, C. Archetti, Z. Feng: *Ants can solve the team orienteering problem*, Computers and Industrial Engineering, Vol. **54** (2008), 648–665. doi:10.1016/j.cie.2007.10.001

[57] S. Lin: *Computer solutions of the traveling salesman problem*, Bell Systems Technology Journal, Vol. **44** (1965), 2245–2269. doi: 10.1002/j.1538-7305.1965.tb04146.x

[58] S. Lin, B. W. Kernighan: *An Effective Heuristic Algorithm for the Traveling-Salesman Problem*, Operations Research, Vol. **21** (1973), 498–516. doi:10.1287/opre.21.2.498

[59] D. Kőnig: *Graphok és mátrixok*, Matematikai és Fizikai Lapok, Vol. **38** (1931), 116–119.

[60] J. B. Kruskal, Jr.: *On the Shortest Spanning Subtree of a Graph and the Travelling Salesman Problem*, in Proceedings of Amererican Mathematics Society, Vol. **7** (1956), 48–50. doi:10.1090/S0002-9939-1956-0078686-7

[61] H. W. Kuhn: *The Hungarian method for the assignment problem*, Naval Research Logistics Quarterly, Vol. **2** (1955), 83–97. doi: 10.1007/978-3-540-68279-0_2

[62] M. Leyzorek, R. S. Gray, A. A. Johnson, W. C. Ladew, S. R. Meaker Jr., R. M. Petry, R. N. Seitz: *Investigation of Model Techniques – A Study of Model Techniques for Communication Systems*, Case Institute of Technology, Cleveland, Ohio (1957).

[63] E. H. C. Lu, C. Y. Lin, V. S. Tseng: *Trip-Mine: An Efficient Trip Planning Approach with Travel Time Constraints*, in Proceedings of the IEEE 12th International Conference on Mobile Data Management, Vol. **1** (2011), 152–161. doi:10.1109/MDM.2011.13

[64] K. Menger: *Ein Theorem über die Bogenlange*, Anzeiger – Akademie der Wissenschaften in Wien – Mathematisch-naturwissenschaftliche, Klasse **65** (1928), 264–266.

[65] C. Miller, A. Tucker, R. Zemlin: *Integer programming formulations and travelling salesman problems*, Journal of the ACM, Vol. **7** (1960), 326–329. doi:10.1145/321043.321046

[66] E. F. Moore: *The shortest path through a maze*, in Proceedings of an International Symposium on the Theory of Switching, 2–5 April 1957, The Annals of the Computation Laboratory of Harvard University, Vol. **30** (1959), Harvard University Press, Cambridge, 285–292.

[67] L. Muyldermans, P. Beullens, D. Cattrysse, D. Van Oudheusden: *Exploring variants of 2- and 3-opt for the general routing problem*, Operations Research, Vol. **53**, No. **6** (2005), 982–995. doi:10.1287/opre.1040.0205

[68] J. Pearl: Heuristics: *Intelligent Search Strategies for Computer Problem Solving*, Addison-Wesley (1984), p. 48. doi:10.1016/S0736-5853(86)80081-8

[69] A. Pessoa, M. Poggi de Aragao, E. Uchoa: *Robust Branch-Cut-and-Price Algorithms for Vehicle Routing Problems*, The Vehicle Routing Problem: Latest Advances and New Challenges, series of the Operations Research/Computer Science Interfaces, Vol. 43 (2008), 297–325. doi: 10.1007/978-0-387-77778-8_14

[70] A. Popescu, G. Grefenstette: *Deducing trip related information from flickr*, in Proceedings of the 18th international conference on World wide web (WWW'2009) (2009), 1183–1184. doi: 10.1145/1526709.1526919

[71] R. C. Prim: *Shortest connection networks and some generalizations*, Bell System Technical Journal, **36** (1957), 1389–1401. doi:10.1002/j.1538-7305.1957.tb01515.x

[72] R. Ramesh, K. Brown: *An efficient four-phase heuristic for the generalized orienteering problem*, Computers and Operations Research, Vol. **18** (1991), 151–165. doi:10.1016/0305-0548(91)90086-7

[73] R. Ramesh, Y. Yoon, M. Karwan: *An optimal algorithm for the orienteering tour problem*, ORSA Journal on Computing, Vol. **4** (1992), 155–165. doi:10.1016/0305-0548(91)90086-7

[74] G. Righini, M. Salani: *Dynamic programming for the orienteering problem with time windows*, Technical Report No. **91** (2006), Dipartimento di Tecnologie dell'Informazione, Universita degli Studi Milano, Crema, Italy. doi:10.1016/j.cor.2008.01.003

[75] L. M. Rousseau, M. Gendreau, G. Pesant, F. Focacci: *Solving VRPTWs with Constraint Programming Based Column Generation*, Kluwer Academic Publishers, Annals of Operations Research, Vol. **130** (2004), 199–216. doi: 10.1023/B:ANOR.0000032576.73681.29

[76] M. Schilde, K. Doerner, R. Hartl, G. Kiechle: *Metaheuristics for the bi- objective orienteering problem*, Swarm Intelligence, Vol. **3** (2009), 179–201. doi: 10.1007/s11721-009-0029-5

[77] A. Schrijver: *Szemelvények a kombinatorikus optimalizálás történetéből (1960-ig)*, Alkalmazott Matematikai Lapok, Vol. **25**, No. **1** (2008), 1–74.

[78] A. Schrijver: *On the History of the Shortest Path Problem*, Documenta Mathematica, Extra Volume ISMP (2012), 155–168. doi:10.1016/S0927-0507(05)12001-5

[79] A. Shimbel: *Structure in communication nets*, in Proceedings of the Symposium on Information Networks (New York, 1954), Polytechnic Press of the Polytechnic Institute of Brooklyn (1955), 199–203.

[80] W. Souffriau, P. Vansteenwegen, J. Vertommen, G. Vanden Berghe, D. Van Oudheusden: *A personalised tourist trip design algorithm for mobile tourist guides*, Applied Artificial Intelligence, Vol. **22**, No. **10** (2008), 964–985. doi: 10.1080/08839510802379626

[81] W. Souffriau, P. Vansteenwegen, G. Vanden Berghe, D. Van Oudheusden: *The planning of cycle trips in the province of East Flanders*, Omega, Vol. **39**, No. **2** (2011), 209–213. doi:10.1016/j.omega.2010.05.001

[82] W. Souffriau, P. Vansteenwegen, G. Vanden Berghe, D. D. Van Oudheusden: *The Multiconstraint Team Orienteering Problem with Multiple Time Windows*, Jounal of Transportation Science, Vol. **47**, No. **1** (2013), 53–63. doi: 10.1287/trsc.1110.0377

[83] K. Sylejmani, A. Dika: *Solving touristic trip planning problem by using taboo search approach*, International Journal of Computer Science Issues, Vol. **8**, Issue **5**, No. **3** (2011), 139–149. doi: 10.1109/HIS.2012.6421351

[84] H. Tang, E. Miller-Hooks: *A tabu search heuristic for the team orienteering problem*, Computer and Operations Research, Vol. **32** (2005), 1379–1407. doi:10.1016/j.cor.2003.11.008

[85] H. Tang, E. Miller-Hooks, R. Tomastik: *Scheduling technicians for planned maintenance of geographically distributed equipment*, Transportation Research, Part E: Logistics and Transportation Review, Vol. **43**, No. **5** (2007), 591–609. doi:10.1016/j.tre.2006.03.004

[86] E. Triantaphyllou: *Multi-Criteria Decision Making: A Comparative Study*, Dordrecht, The Netherlands: Kluwer Academic Publishers (now Springer) (2000), p. 320. doi:10.1007/978-1-4757-3157-6

[87] F. Tricoire, M. Romauch, K. Doerner, R. Hartl: *Heuristics for the multi-period orienteering problem with multiple time windows*, Computers and Operations Research, Vol. **37**, No. **2** (2010), 351–367. doi:10.1016/j.cor.2009.05.012

[88] T. Tsiligirides: *Heuristic methods applied to orienteering*, Journal of the Operational Research Society, Vol. **35**, No. **9** (1984), 797–809. doi:10.1057/jors.1984.162

[89] P. Vansteenwegen, D. Van Oudheusden: *The mobile tourist guide: An or opportunity*, OR Insights, Vol. **20**, No. **3** (2007), 21–27. doi:10.1057/ori.2007.17

[90] P. Vansteenwegen, W. Souffriau, G. Vanden Berghe, D. Van Oudheusden: *A guided local search metaheuristic for the team orienteering problem*, European Journal of Operational Research, Vol. **196**, No. **1** (2009), 118–127. doi:10.1016/j.ejor.2008.02.037

[91] P. Vansteenwegen, W. Souffriau, G. Vanden Berghe, D. Van Oudheusden: *Iterated local search for the team orienteering problem with time windows*, Computers and Operations Research, Vol. **36**, No. **12** (2009), 3281–3290. doi:10.1016/j.cor.2009.03.008

[92] P. Vansteenwegen, W. Souffriau, G. Vanden Berghe, D. Van Oudheusden: *Metaheuristics for tourist trip planning*, In: M. Geiger, W. Habenicht, M. Sevaux, K. Sörensen (eds.): Metaheuristics in the Service Industry, Lecture Notes in Economics and Mathematical Systems, Vol. **624** (2009), 15–31. doi:10.1016/j.cor.2015.03.016

[93] P. Vansteenwegen, W. Souffriau, D. Van Oudheusden: *The orienteering problem: a survey*, European Journal of Operational Research, Vol. **209**, No. **1** (2011), 1–10. doi:10.1016/j.ejor.2010.03.045

[94] C. Verbeeck, K. Sörsensen, E. H. Aghezzaf, P. Vansteenwegen: *A fast solution method for the time-dependent orienteering problem*, European Journal of Operational Research, Vol. **236** (2014), 419–432. doi:10.1016/j.ejor.2013.11.038

[95] X. Wang, B. Golden, E. Wasil: *Using a genetic algorithm to solve the generalized orienteering problem*, In: B. Golden, S. Raghavan, E. Wasil (Eds.): The Vehicle Routing Problem: Latest Advances and New Challenges (2008), 263–274. doi:10.1007/978-0-387-77778-8_12

[96] C. Wiener: *Ueber eine Aufgabe aus der Geometria situs*, Mathematische Annalen Vol. 6 (1873), 29–30.

[97] A. Yahi, A. Chassang, L. Raynaud, H. Duthil, D. H. Chau: *Aurigo - An Interactive Tour Planner for Personalized Itineraries*, in Proceedings of the 20th International Conference on Intelligent User Interfaces, 275–285. doi: 10.1145/2678025.2701366

[98] http://www.dis.uniroma1.it/challenge9/format.shtml

A.1. függelék: Az Orienteering Problem formalizálása

Legyen adott egy $G(V,E)$ gráf, amelynek minden v_i csúcsához egy π_i nemnegatív profitérték van rendelve, melyet az ügynök megkap, ha meglátogatja a v_i csúcsot, valamint v_i és v_j csúcsok közötti e_{ij} élhez t_{ij} élköltséget rendelünk, ami a távolság megtételéhez szükséges idő. A feladat T_{max} idő alatt maximális pontot összegyűjteni úgy, hogy minden csúcs legfeljebb egyszer látogatható meg. A kezdő- és a végpont fix, és gyakran meg is egyeznek egymással. Jelölje továbbá h_i, hogy az i-edik csúcs hanyadik lépésben kerül sorra az úton, valamint τ_{ij} értéke legyen 1, ha az i-edik csúcs után a j-edik következik az úton, és 0 különben. Ugyan fontos szerepet játszik az egyes csúcsok kiválasztásában az ott töltendő idő is, ám ezt gyakran nem szerepeltetik a modellben, inkább szétosztják a csúcs előtti és utáni élekre (jellemzően fele-fele arányban). Ekkor az OP formalizálása a következőképpen alakul:

$$\max \sum_{i=2}^{N-1} \sum_{j=2}^{N} \pi_i \tau_{ij}$$

$$\sum_{j=2}^{N} \tau_{1j} = \sum_{i=1}^{N-1} \tau_{iN} = 1$$

$$\sum_{j=2}^{N} \tau_{kj} = \sum_{i=1}^{N-1} \tau_{ik} \leqslant 1; \qquad \forall k = 2, \ldots, N-1$$

$$\sum_{i=1}^{N-1} \sum_{j=2}^{N} \tau_{ij} t_{ij} \leqslant T_{max}$$

$$h_i - h_j + 1 \leqslant (N-1)(1-\tau_{ij}); \qquad \forall i,j = 2, \ldots, N$$

$$2 \leqslant h_i \leqslant N; \qquad \forall i = 2, \ldots, N$$

$$\tau_{ij} \in \{0,1\} \quad \forall i,j = 1, \ldots, N$$

Az egyes sorok jelentése a következő:

1. A célfüggvény: a csúcsoknál begyűjtött profitok összege legyen maximális.
2. Az út az 1-es csúcsnál kezdődik, és az N-ediknél ér véget.
3. Az út összefüggő, és minden csúcsot csak legfeljebb egyszer látogatunk meg.
4. Betartjuk az időkorlátot.
5. és 6. együtt garantálja, hogy ne legyenek körök az útban, Miller-Tucker-Zemlin javaslata alapján [65].
7. A τ_{ij} értékkészlete 0 vagy 1.

A.2. függelék: Az Orienteering Problem with Time Windows formalizálása

Az OP-nél leírtaktól annyiban tér el az OPTW, hogy minden csúcsot csak az $[O_i, C_i]$ nyitvatartási ideje alatt lehet meglátogatni, és jelöljük s_i-vel az i-edik csúcshoz való megérkezés időpontját. Ekkor az OPTW leírható az alábbi módon:

$$\max \sum_{i=2}^{N-1} \sum_{j=2}^{N} \pi_i \tau_{ij}$$

$$\sum_{j=2}^{N} \tau_{1j} = \sum_{i=1}^{N-1} \tau_{iN} = 1$$

$$\sum_{j=2}^{N} \tau_{kj} = \sum_{i=1}^{N-1} \tau_{ik} \leqslant 1; \quad \forall k = 2, \ldots, N-1$$

$$\sum_{i=1}^{N-1} \sum_{j=2}^{N} \tau_{ij} t_{ij} \leqslant T_{max}$$

$$s_i + t_{ij} - s_j + 1 \leqslant M(1 - \tau_{ij}); \quad \forall i, j = 1, \ldots, N$$

$$O_i \leqslant s_i \leqslant C_i; \quad \forall i = 1, \ldots, N$$

$$\tau_{ij} \in \{0, 1\} \quad \forall i, j = 1, \ldots, N$$

Látható, hogy az OP-hez képest csupán a körmentesség feltétele változott (itt M egy nagy konstans értéket jelöl), valamint kibővült a nyitvatartási idő korlátjával a feltételrendszer.

A.3. függelék: A Team Orienteering Problem formalizálása

Legyen adott egy $G(V, E)$ gráf, amelynek minden v_i csúcsához egy i nemnegatív profitérték van rendelve, melyet az ügynök megkap, ha meglátogatja a v_i csúcsot, valamint v_i és v_j csúcsok közötti e_{ij} élhez t_{ij} élköltséget rendelünk, ami a távolság megtételéhez szükséges idő. A feladat T_{max} idő alatt P darab ügynök számára maximális pontot összegyűjteni úgy, hogy minden csúcs legfeljebb egyszer látogatható meg. A kezdő- és a végpont fix, és gyakran meg is egyeznek egymással. Jelölje továbbá h_{ip}, hogy a p-edik útnál az i-edik csúcs hanyadik lépésben kerül sorra az úton, valamint τ_{ijp} értéke legyen 1, ha a p-edik útnál az i-edik csúcs után a j-edik következik az úton, és 0 különben. Legyen $\Theta_i p$ értéke 1, ha a p-edik úton az i-edik csúcsot meglátogatják, és 0 különben. Ekkor a TOP megfogalmazható a következőképpen:

$$max \sum_{p=1}^{P} \sum_{i=2}^{N-1} \pi_i \theta_{ip}$$

$$\sum_{p=1}^{P} \sum_{j=2}^{N} \tau_{1jp} = \sum_{p=1}^{P} \sum_{i=1}^{N-1} \tau_{iNp} = P$$

$$\sum_{p=1}^{P} \theta_{kp} \leqslant 1; \quad \forall k = 2, \ldots, N-1$$

$$\sum_{j=2}^{N} \tau_{kjp} = \sum_{i=1}^{N-1} \tau_{ikp} = \theta_{kp}; \quad \forall k = 2, \ldots, N-1; \quad \forall p = 1, \ldots, P$$

$$\sum_{i=1}^{N-1} \sum_{j=2}^{N} \tau_{ijp} t_{ij} \leqslant T_{max}; \quad \forall p = 1, \ldots, P$$

$$h_{ip} - h_{jp} + 1 \leqslant (N-1)(1 - \tau_{ijp}); \quad \forall i, j = 2, \ldots, N; \quad \forall p = 1, \ldots, P$$

$$2 \leqslant h_{ip} \leqslant N; \quad \forall i = 2, \ldots, N; \quad \forall p = 1, \ldots, P$$

$$\tau_{ijp}, \theta_{ip} \in \{0, 1\} \quad \forall i, j = 1, \ldots, N; \quad \forall p = 1, \ldots, P$$

Az egyes sorok jelentése a következő:

1. A célfüggvény: a csúcsoknál begyűjtött profitok összege legyen maximális az összes utat figyelembe véve.
2. Minden út az 1-es csúcsnál kezdődik, és az N-ediknél ér véget.
3. Minden csúcsot csak legfeljebb egyszer látogatunk meg.
4. Minden út egyenként összefüggő.
5. Betartjuk az időkorlátot.
6. és 7. együtt garantálja, hogy ne legyenek körök az útban, Miller-Tucker-Zemlin javaslata alapján [65].
8. A τ_{ijp} és Θ_{ip} értékkészlete 0 vagy 1.

A.4. függelék: A Team Orienteering Problem with Time Windows formalizálása

A TOP-nél leírtaktól annyiban tér el a TOPTW, hogy minden csúcsot csak az $[O_i, C_i]$ nyitvatartási ideje alatt lehet meglátogatni, és jelöljük s_{ip}-vel a p-edik út során az i-edik csúcshoz történő megérkezés időpontját. Ekkor a TOPTW leírható az alábbi módon:

$$\max \sum_{p=1}^{P} \sum_{i=2}^{N-1} \pi_i \theta_{ip}$$

$$\sum_{p=1}^{P} \sum_{j=2}^{N} \tau_{1jp} = \sum_{p=1}^{P} \sum_{i=1}^{N-1} \tau_{iNp} = P$$

$$\sum_{p=1}^{P} \theta_{kp} \leqslant 1; \quad \forall k = 2, \ldots, N-1$$

$$\sum_{j=2}^{N} \tau_{kjp} = \sum_{i=1}^{N-1} \tau_{ikp} = \theta_{kp}; \quad \forall k = 2, \ldots, N-1; \quad \forall p = 1, \ldots, P$$

$$\sum_{i=1}^{N-1} \sum_{j=2}^{N} \tau_{ijp} t_{ij} \leqslant T_{max}; \quad \forall p = 1, \ldots, P$$

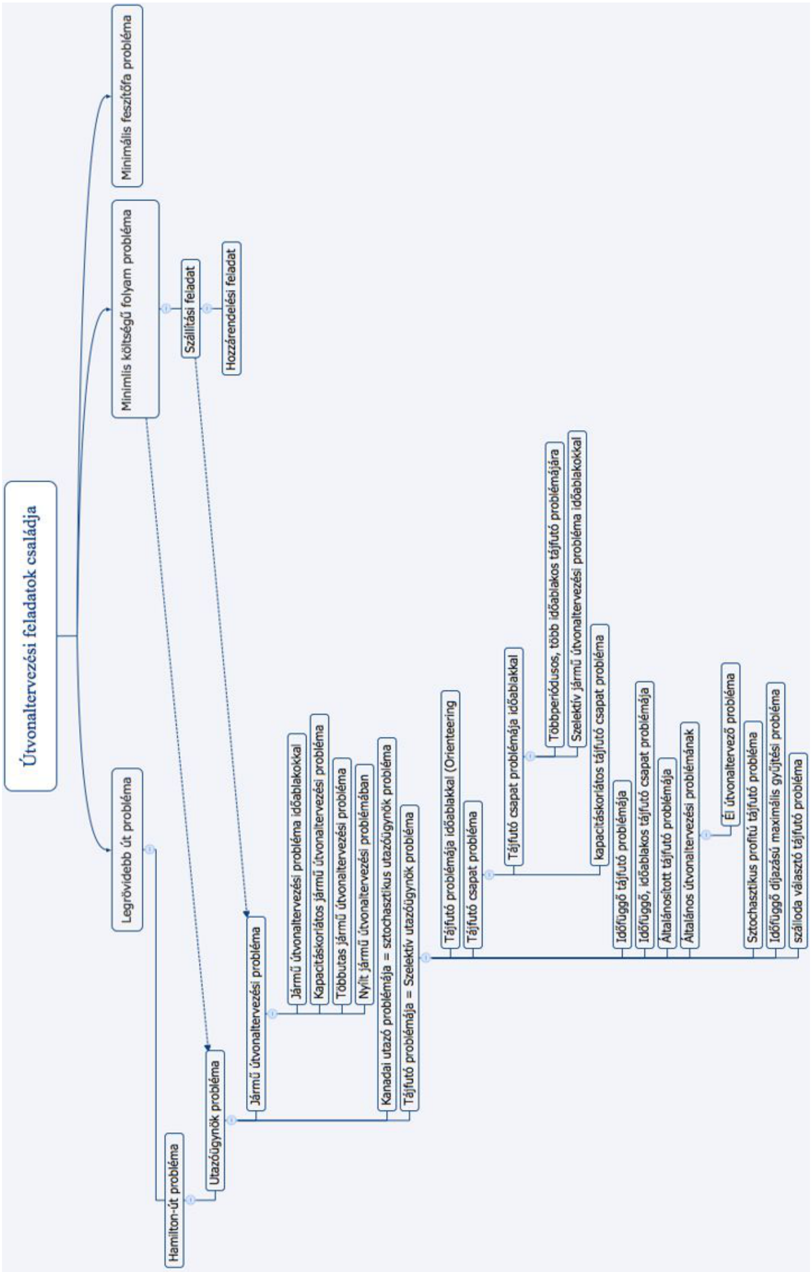
$$s_{ip} + t_{ij} - s_{jp} \leqslant M(1 - \tau_{ijp}); \quad \forall i, j = 1, \ldots, N; \quad \forall p = 1, \ldots, P$$

$$O_i \leqslant s_{ip} \leqslant C_i; \quad \forall i = 1, \ldots, N; \quad \forall p = 1, \ldots, P$$

$$\tau_{ijp}, \theta_{ip} \in \{0, 1\} \quad \forall i, j = 1, \ldots, N; \quad \forall p = 1, \ldots, P$$

Látható, hogy az OP-hez képest csupán a körmentesség feltétele változott (itt M egy nagy konstans értéket jelöl), valamint kibővült a nyitvatartási idő korlátjával a feltételrendszer.

B függelék: Az útvonaltervező feladatok családja



(Beérkezett: 2016. június 1.)

APÁTHY M. SÁNDOR
Budapesti Corvinus Egyetem, Matematika Tanszék
Lakcím: 1064 Budapest, Vörösmarty utca 25/A
sandor.m.apathy@gmail.com

A SURVEY ON ROUTE PLANNING ALGORITHMS, FOCUSING ON ITS TOURISTIC APPLICATIONS

Sándor M. Apáthy

There are few worse situations I could imagine than getting lost in a foreign city. It is even worse if the language barriers keeps us away from the chance of getting help. These experiences might have inspired many Researchers on the field of Route planning algorithms. The aim of this paper is to briefly present the cumbersome research efforts that lead to the recent touristic and transportation related algorithms. We start our survey from the Shortest Path Problem to show how the wide range of route planning problems were unified by the technique of Linear Programming, then a far-reaching set of transportation and travelling problems will be introduced that unfolded from the Traveling Salesman Problem. To keep the extent of the survey at a reasonable level our focus is narrowed more to the Touristic solutions due to the growing number of specific routing methods from the early '60s.

Keywords: Team Orienteering Problem, Route Planning, Heuristic Algorithm, Tourism JEL code: C60, C61, Z32