



**Közgazdaságtudományi
Doktori Iskola**

TÉZISGYŰJTEMÉNY

Apáthy M. Sándor

Egy turisztikai útvonaltervező ajánlórendszer modellje

című Ph.D. értekezéséhez

Témavezető:

Tallos Péter, CSc
egyetemi tanár

Budapest, 2016

**Budapesti Corvinus Egyetem
Matematika Tanszék**

TÉZISGYŰJTEMÉNY

Apáthy M. Sándor

Egy turisztikai útvonaltervező ajánlórendszer modellje

című Ph.D. értekezéséhez

Témavezető:

Tallos Péter, CSc
egyetemi tanár

© Apáthy M. Sándor

Tartalomjegyzék

1. Kutatási előzmények és a téma indoklása	4
2. Felhasznált módszerek	9
2.1. Személyre szabott menetidőbecslés	9
2.2. Turisztikai ajánlórendszer	15
2.3. Útvonaltervezés	20
3. A kutatási eredmények összegzése	31
4. Hivatkozások	35
5. Saját publikációk a témakörben	37

1. Kutatási előzmények és a téma indoklása

Kevés rémesebb érzést tudok elképzelni, mint eltévedni egy idegen városban, főként, ha nyelvi akadályok miatt még segítségkérésre sem igen van lehetőségünk. Bár ez velem is megtörtént több alkalommal, sokkal inkább az ösztökél turisztikai ajánlórendszer tervezését célzó kutatásaim során, hogy az újdonság különleges és lebilincselő érzésének megélésében segítsék másokat. Abban, mikor egy addig számukra ismeretlen kultúrával találkoznak, vagy egy addig még sosem látott építészeti megoldással egy ház homlokzatán. Nem titkolt célom, hogy jelen kutatásra alapozva a későbbiekben mobil alkalmazás formájában is megmértessem eredményeim. Képtelen lennék elfogadni, hogy olyan kutatásba fektessenek energiát, mely nem bír gyakorlati hasznossággal. Amennyiben egy majdani mobil alkalmazás akár csak egy felhasználót is hozzásegít egy olyan - pozitív - élményhez, mely annak hiányában elkerülte volna, már megérte a munka. Az alábbiakban a kutatás aktuális állapota kerül bemutatásra.

Fontos megemlíteni, hogy a dolgozat néhány kivételtől eltekintve többszám első személyben íródott, de ez nem társszerzőkre utal, csupán a szerző stilisztikai döntése.

A kutatás célja

Életem során megannyi várost volt szerencsém bebarangolni, ám gyakran jelentett komoly kihívást olyan túrák megtervezése, melyek a rendelkezésemre álló idő minél hatékonyabb kihasználását teszik lehetővé. **Jelen dolgozat célja egy olyan turisztikai ajánlórendszer elméleti alapjainak lefektetése, mely képes elegendően pontos és személyre szabott útvonalak tervezésére egy idegen városban, figyelembe véve a felhasználó igényeit és lehetőségeit.** Ezen igények és lehetőségek megismerése, és modellbe történő beépítése nem magától értetődő feladat, a dolgozat szerzője azonban erre tesz kísérletet. Elsőként a turisták sebességének becslését célozzuk, adott körülményeket (mint például az út meredeksége, vagy épp pillanatnyi fizikai erőnléte) figyelembe véve, mellyel személyre szabottan tudjuk két helyszín közötti menetidejét becsülni. Ezt követően egy olyan modellt építünk, mely néhány információ alapján igyekszik a felhasználó ízlésvilágáról minél pontosabb képet alkotni, az ajánlórendszerek köréből származó módszerek segítségével. A továbbiakban ez lehet segítségünkre abban, hogy megértsük, a túra során relatíve mennyire értékelné egy adott

helyszínt, vagy nevezetességet az útvonalba beillesztve. Amennyiben a fenti információk már a rendelkezésünkre állnak, **lehetőségünk nyílik egy olyan útvonaltervező algoritmus megalkotására, mely ezeket felhasználva a túrázó igényeihez mérten leginkább testreszabott túrát tud ajánlani egy adott városban.**

A tézis során az alábbi kérdéskörökre igyekszünk választ találni:

1. Hogyan lehet a GPS alapú túranaplók adatait pontosabbá és elemzésre alkalmassá tenni?
Létezik olyan digitális emelkedési modell (DEM), mely a valós magasság értékeket jól közelíti?

A GPS eszközök pontatlanságából adódóan korrekcióra szorulnak a szélességi- és hosszúsági értékek, valamint a magassági adatok. Előbbit az adatpontok korrigálásával, valamint Kálmán-filter segítségével pontosítjuk (2.5.1-es alfejezet), utóbbit két általunk alkotott digitális földfelszín modell bevezetésével (2.4-es alfejezet).

2. Létezik a sebesség és az adott szakasz meredeksége közötti összefüggést leíró ismert megoldásoknál pontosabb?

A gyalogos menetidőbecslő eljárások szakirodalma igen szűkös, melynek vélhetően egyik legfőbb oka, hogy igen nehéz jó minőségű túranaplókhoz hozzájutni. A tanulmányozott szakirodalom alapján kijelenthetjük, hogy jelen dolgozatban az eddigi legnagyobb adathalmazon végezzük a meredekség és sebesség közötti összefüggés becslését (2.5.3-as alfejezet), mely a korábbiaknál nagyobb pontosságot eredményez.

3. Milyen eljárással adható a túrázók menetidejének egy a jelenlegieknél pontosabb becslése?

Jelen dolgozatban két menetidőbecslő eljárás is bemutatásra kerül, melyek pontossága jóval túlszárnyalja az eddig ismert megoldásokat. Az első eljárás a korábban megismert meredekség és sebesség közötti becsült összefüggésre alapoz, melyet fittséget leíró korrekciós tényezővel pontosítunk, míg a második eljárás a korábban tapasztalt sebességekre alapozza becslését. A két eljárást a 2.5.4-es alfejezetben ismertetjük.

4. Hogyan lehet kevés kezdeti információ alapján turisztikai célú ajánlórendszert építeni, mely személyre szabott ajánlásokat tesz a felhasználónak?

Annak érdekében, hogy minimális kezdeti információ felhasználásával tudjunk ajánlásokat tenni, 17 turisztikai faktort határoztunk meg, melyek a látványosságokat hivatottak leírni, továbbá az empirikus vizsgálathoz létrehoztunk egy 3 város látványosságait felölelő adatbázist, melyben klasszifikáljuk a látványosságokat a fent említett faktorok segítségével (3.7-es alfejezet). Az így megalkotott hibrid ajánlórendszer néhány kezdeti kérdésre adott válasz alapján képes ajánlásokat tenni.

5. Lehetséges a turistákat klasszifikálni a kezdetben magukkal kapcsolatban megadott információk alapján? Hogyan építhető fel ennek érdekében egy személyre szabott ajánlásokat adó rendszer kérdőíve?

Empirikus vizsgálatunk során sort kerítettünk a kérdőív kitöltőinek szegmentálására (3.8-as alfejezet), mely alapján szignifikánsan jobb ajánlásokat tudunk adni, mint annak hiányában. A kérdőívben kulcs szerepet játszik a 17 turisztikai faktor meghatározása (3.7.6-os alfejezet).

6. Milyen hasznosságfüggvénnyel írható le általános módon a felhasználók látványosságokra vonatkozó értékelése? Milyen célfüggvény vezet a gyakorlatban a felhasználók számára leginkább elfogadható útvonaltervezéshez?

Az utazóügynök probléma [1] óta minden útvonaltervező feladat célfüggvényét a gráf csúcsaiban gyűjtött profitok összegeként definiálják. Tesztjeink során ez a felhasználók elégedettsége szempontjából kifejezetten gyenge eredményekre vezetett, így egy minden korrábitól eltérő hasznossági- és célfüggvény került meghatározásra (4.4.2-es alfejezet), melynek célja a felhasználói igények minél pontosabb leírása.

7. Milyen algoritmus adható a kitűzött útvonaltervezési feladatra, mely alacsony futási idejével alkalmazhatóvá teszi azt egy valós applikációban?

Az általunk kitűzött útvonaltervezési feladatra egy olyan heurisztikus algoritmust adtunk megoldásként, mely első lépésként lecsökkenti a feladat méretét, majd tovább egyszerűsíti azt klaszterek kialakításával. Ezek segítségével még a viszonylag magas számításigényű optimalizáló szakasz ellenére is rövid futási időt sikerült elérni. Az algoritmust a 4.4.4-es alfejezetben ismertetjük.

A tézis felépítése

Jelen dolgozat a fentiekben megismert kutatási célokat követve hármas tagolású.

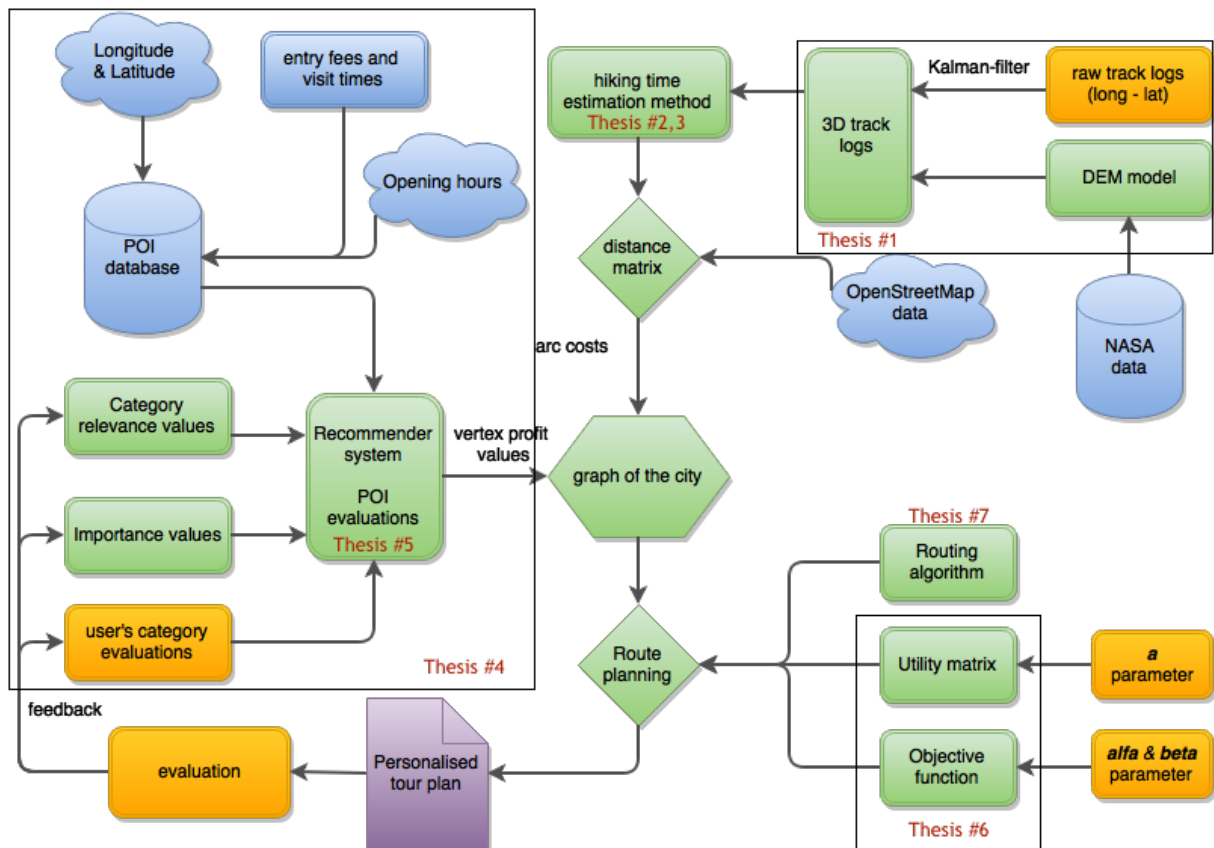
A 2. fejezetben kísérletet teszünk a menetidőbecslés modellezésére, melyhez a szakirodalom alapos tanulmányozása során sem tapasztalt mennyiségű túranapló adatot használtunk fel. A túranaplók előkészítéséhez szükséges megbízható magasságadatok becslésére egy saját földfelszín közelítő modellt építünk NASA adatokra alapozva, majd az így nyert magasság értékeket rendeljük a meglévő szélességi és hosszúsági adatokhoz. Ezután Tobler korábbi munkája nyomán újrabecsüljük a sebességet az útszakasz meredekségével leíró függvényét, és ezt használjuk fel személyre szabott menetidő becslésre, illetve egy másik, átlagsebességeken alapuló eljárást is bemutatunk.

A turisták igényeinek személyre szabott kielégítése érdekében képesnek kell lennünk a preferenciáik feltérképezésére, és ezt figyelembe véve megállapítani az egyes helyszínek meglátogatásához kapcsolódó relatív profitokat. Ennek érdekében a 3. fejezetben bemutatásra kerül az ajánlórendszerek széles szakirodalma, valamint néhány jelenleg használatos turisztikai ajánlórendszer, és ezeket figyelembe véve alkotjuk meg saját ajánlások adására alkalmas hibrid modellünket. Ez egyszerre épít a felhasználóktól kapott információkra, mellyel feltérképezhetjük preferenciáikat, illetve látványosságokra, mint termékek komplex struktúrájára, mely segít megérteni azok szerkezetét. Az empirikus vizsgálat során gyűjtött információk segítségével módunk nyílik a turisták típusainak meghatározására, mely lehetőséget ad az ajánlások pontosítására.

A 4. fejezetben egy, a turista igényeihez alkalmazkodni képes útvonaltervező algoritmust alkotunk meg. A modellezés során a várost egy N csúcsú irányítatlan gráffal jellemzünk, melynek minden csúcsa egy-egy potenciálisan meglátogatandó látványosságot jelképez, míg az élek a végpontjaikat összekötő legrövidebb utat hivatottak leírni. Az egyes csúcsokban begyűjthető profitokat a 3. fejezetben megalkotott ajánlórendszerünkől származtatjuk, így azok speciálisan az adott turista igényeihez igazodnak. A gráf élköltségei nem mások, mint a turista számára az adott két pont között vezető legrövidebb út menetideje. A feladat, hogy P nap alatt (napi T órában) olyan uta(ka)t járjon be a gráfon, mellyel a célfüggvénye értékét maximalizálja. Külön figyelmet szenteltünk egy olyan célfüggvény megalkotásának, mely merőben eltér az útvonaltervező algoritmusok gyakorlatától annak érdekében, hogy a

felhasználók számára leginkább tetsző útvonalat legyünk képesek tervezni. Az 1. ábrán foglaltuk össze a tervezés során használt adatforrásokat (ezeket kékkel jelöltük, és felhő alakú, amennyiben adatbányász eljárással jutottunk hozzá az internetről), a narancs színnel jelöltük a felhasználótól származó bemeneti adatokat, és zölddel az általunk kalkulált elemeket.

1. ábra: A turisztikai ajánlórendszer logikai felépítése



Ahogy az látható, a várost leképező gráfhhoz három adat szükséges, a látványosságok helyzeti adatai, amit az OpenStreetMap alkalmazásból nyerünk, az élköltségek, amit a távolságmátrix szolgáltat, valamint a csúcsokban gyűjthető profitok, melyeket az ajánlórendszerből nyerünk. A távolságmátrix a menetidőbecslő eljárásan alapszik, melynek modellezése során a turautak.hu-tól szerzett túranaplókat használtuk fel a saját digitális földfelszín modellünk mellett (ami NASA adatokon alapszik). Az ajánlórendszerhez felhasználjuk az általunk épített adatbázist a látványosságokkal és azok attribútumaival (úgy mint helyzeti adatok, látogatási- és nyitvatartási idők, valamint belépődíjak), továbbá a felhasználók 17 kategóriára tett értékelését és a látványosságok kategóriákra vonatkozó relevanciaértékeit és “fontossági” paramétereit. Az így előálló gráfon tudjuk alkalmazni útvonaltervező algoritmusunkat, melynek kulcs elemei a felhasználók által adott paraméterek

(pl. lustaság), és az általunk elkészített hasznossági- és célfüggvény, melyet a feladat során maximalizálni igyekszünk. Az algoritmus generálta személyre szabott útvonaltervre adott értékelés folyamatos visszacsatolást ad a rendszernek, mely tovább pontosítja az ajánlásokat. Az 5. fejezetben a kutatási eredményeket foglaljuk össze, míg a kutatás lehetséges jövőbeli irányai témakörönként az adott fejezetek végén kerülnek kijelölésre.

2. Felhasznált módszerek

2.1. Személyre szabott menetidőbecslés

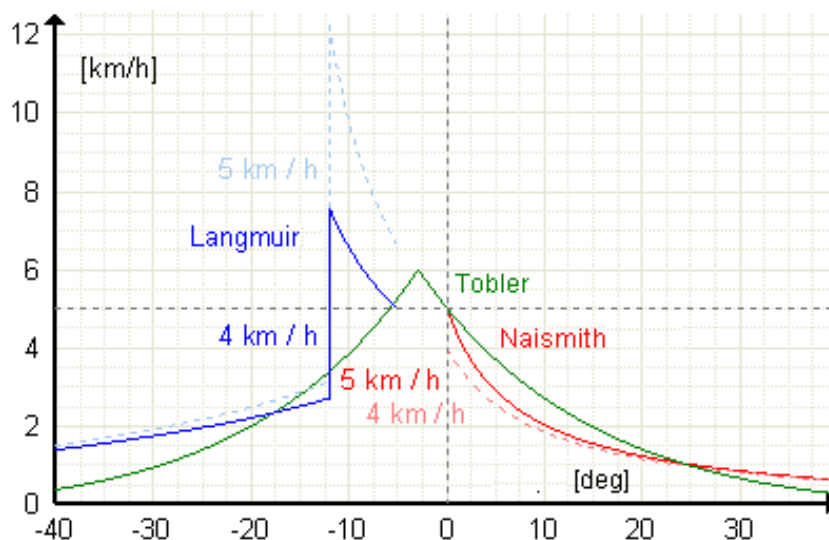
A túrautakra vonatkozó menetidő becslés William Naismith, skót hegymászó 1892-ben meghatározott ökölszabályával kezdődött [3], mely szerint 1 óra alatt 3 mérföldet (4827,9 méter) tud megtenni egy “átlagos” kondícióval bíró személy, “tipikus” terepviszonyok mellett és “normál” körülményeket feltételezve (hőmérséklet, páratartalom, szél, stb.), míg minden 2000 láb (632 méter) emelkedő további 1 órát vesz igénybe. A gyakorlatban tehát a sík terepen és emelkedőn való mozgás ekvivalenciáját mondja ki, vagyis 1 egység emelkedő 7,92 egység sík terepen megtett távolsággal egyenlő idő alatt teljesíthető (ezt szokás 1:8 szabályként is emlegetni).

Az idők folyamán megannyi módosítási javaslat született:

- Aitken [5] feltevése szerint úton és ösvényen elfogadható a Naismith-szabály, de minden egyéb felüleleten 20%-kal gyengébben teljesít a túrázó.
- Langmuir [4] Naismith becslését ambíciózusnak tartotta, és 4km/h sebességet feltételezett sík terepen (+-5 fok eltérés esetén), továbbá javasolja, hogy minden 300m-en csökkentsük a becsült menetidőt enyhe lejtőn (5-12 fok között) és növeljük 10 perccel minden 300m-en meredek lejtőn (12 foknál nagyobb).
- Tobler a gyaloglás sebességét exponenciális függvénnyel becsülte az út meredekségének függvényében [6]. Ennek maximuma kb. 6km/h kis meredekségű lejtőn, míg a sebesség 0-hoz közelít +-60 fok esetén, tehát extrém meredekségű emelkedőn vagy lejtőn.

A különféle becsléseket az 2. ábrán foglaljuk össze, ahol a becsült sebességet láthatjuk a meredekség (fokban mérve) függvényében.

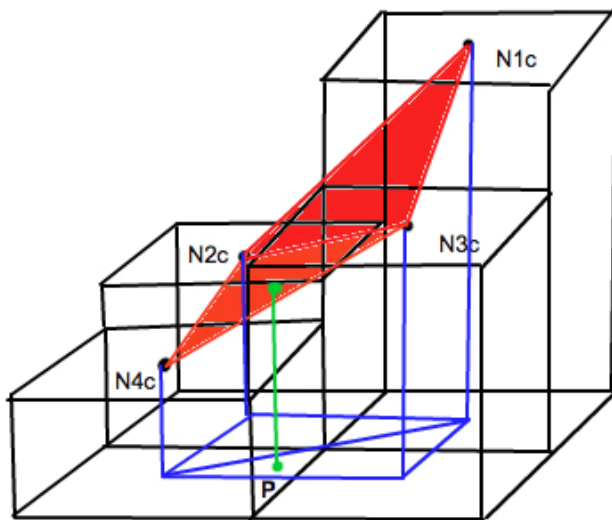
2. ábra: Becslési eljárások összevetése



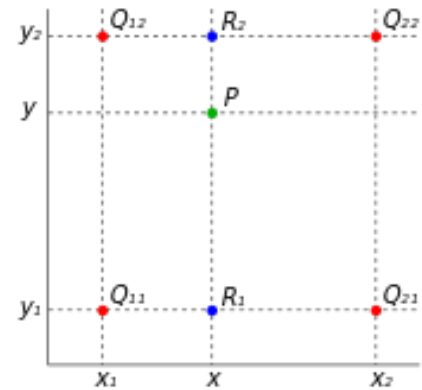
Földfelszín modell

A felhasznált magasságs adatok a NASA által nemrégiben közzétett adatokon alapszanak [13]. Ez lényegében 30×30 méteres négyzetekre osztja a földfelszínt, és ezekhez rendel magasság értékeket. Az általunk alkalmazott földfelszín közelítés a bilineáris interpoláció elvén alapszik, ahol a 4 egymás melletti négyzet alapú hasáb fedlapjainak középpontjaira számolt átlagokkal közelítjük a valódi felszínt (lásd a 3. ábrán). Ekkor a felszínen adott P pont magasság koordinátáját úgy számoljuk, hogy a négyzetek felszínre eső merőleges vetületeivel vett téglalapok területének arányában súlyozzuk a megfelelő középpontokhoz tartozó magasság értékeket (lásd a 4. ábrán). Fontos megemlíteni, hogy az $(N_{1,c}, N_{2,c}, N_{3,c}, N_{4,c})$ csúcsok a legritkább esetben esnek egy síkba, így a földfelszínt sem az általuk kifeszített “négyzetekkel” közelítjük, hanem a helyvektoraik konvex kombinációjaként előálló vektorok összességével. Ennek értelmében a számított pont is csak véletlenül eshet $(N_{1,c}, N_{2,c}, N_{3,c}, N_{4,c})$ csúcsok közül bármelyik három által kifeszített síkba. A pontos számítást a B mellékletben ismertetjük. Az így számított magasságs adatokat ismét összevetettük a Google alkalmazás által kalkulált értékekkel a korábbihoz hasonló módon, és kevesebb, mint 1 cm-en belüli eltérést tapasztaltunk, mely sejteti, hogy a Google is hasonló elvek mentén és hasonló alapadatokból (NASA, [13]) készítette földfelszín modelljét, mint a mi “négyzet-modellünk”

3. ábra: Négyzög-modell



4. ábra: Bilineáris interpoláció



A sebesség becslése a meredekség függvényében

Mi a rendelkezésünkre álló kb. 2.400 túranapló alapján kívánjuk becsülni a túrázó sebessége és a terepszakasz meredeksége közötti összefüggést.

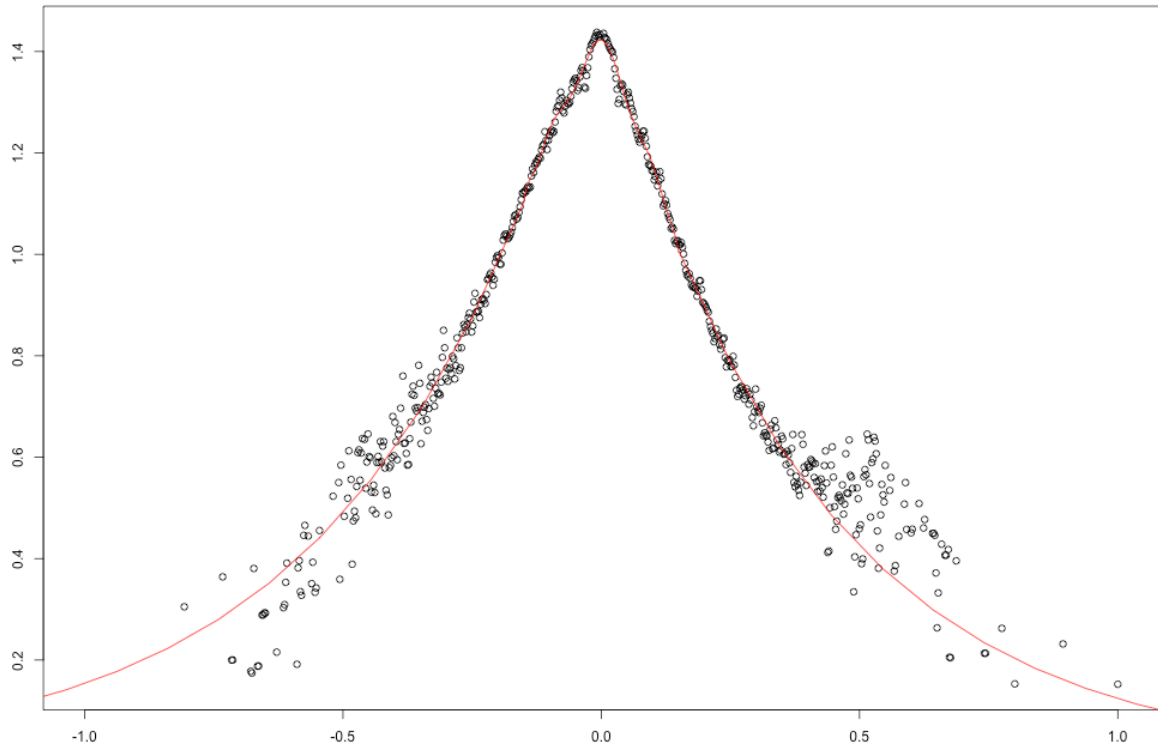
Az egyes szakaszokra vonatkozó sebesség-meredekség párokat tekintve (az outlierektől a korábbiakban leírt módszerrel való megtisztítás után) 1/4 fokként haladva a meredekségi adatokon, az adott érték körüli $\pm 0,125$ fokos intervallumban található sebességek számtani átlagát véve számolunk egy átlagos sebesség értéket minden negyed fokhoz a meredekség skálán. Az illesztett görbe a $[-0,15; 0,15]$ intervallumon egy 10-ed fokú polinom, míg a széleken egy-egy exponenciális függvényt illesztettünk az átlagokra. Ennek legfőbb oka, hogy korábbi kutatási eredményeink alapján a Tobler-görbe maximum pont körül rosszul illeszkedik valós adatokra, így azt minél inkább igyekeztük lekövetni egy polinommal, másrészt a polinom a széleken rendszerint rosszul illeszkedik, ezért a gyakorlatnak sokkal inkább megfelelő (a széleken a vízszintes tengelyhez simuló) görbét kerestünk.

Az illesztés eredményeként az alábbi sebességet becsülő függvényt kaptuk:

$$v(m) = \begin{cases} e^{2.3203m+0.4462} & | m \in (-\infty; -0,15) \\ p(m) & | m \in [-0,15; 0,15] \\ e^{-2.4672m+0.3769} & | m \in (0,15; \infty) \end{cases}$$

ahol m az adott szakasz meredekségét jelöli. A sebességet a meredekség függvényében becsülő $v(m)$ görbénket a 5. ábrán láthatjuk.

**5.ábra: Az átlagsebességekre illesztett becslés
(meredekség radiánban, sebesség m/s-ban mérve)**



Ezt alapul véve a jelen szakaszban két menetidőbecslő eljárást ismertetünk, melyet a korábban bemutatott 2400 túranaplón teszteltünk.

1. Meredekség alapú eljárás

A korábbiakban bemutatott eljárás szerint illesztünk sebességet becsülő görbét a tesztalmezben szereplő túranaplók szakaszainak meredekség-sebesség párojaira, majd ezt személyre szabjuk az alábbiak szerint: A túraút első 20 másodperces szakaszának meredeksége legyen m_1 , ekkor ennek sebességét becsüljük az illesztett $v(m)$ görbe alapján $v(m_1)$ -gyel. A második szakasz sebességének becsülésénél már felhasználjuk, hogy az előző szakasz becsült értékét össze tudjuk hasonlítani a valós adattal (túra közben). A valós és becsült sebesség arányát tekinthetjük ezt egy fittségi faktornak, mely adott meredekség mellett az átlagos túrázó sebességétől vett eltérését mutatja. Legyen ennek értéke $b_1 = v_1 / v(m_1)$. Ekkor a második szakasz becsült sebessége legyen $b_1 v(m_2)$, ahol m_2 a második 20

másodperces szakasz meredeksége. A harmadik szakasz sebességének becslésekor már felhasználjuk a 2. szakasz megfigyelt fittségi faktorát is, és vesszük a számtani átlagukat, tehát a becsült sebessége $(b_1+b_2)v(m_3)/2$ lesz. Általánosan az n-edik szakasz sebességének a becslése az alábbiak szerint történik:

$$v_n^* = \frac{\sum_{i=1}^n b_{i-1} v(m_i)}{n-1}$$

Ezzel az útközben történő kiigazítással a teljes úthosszra tett becslést jelentősen javíthatjuk, hiszen olyan befolyásoló körülményeket tudunk részben leképezni, mint az időjárási viszonyok, vagy a túrázó aktuális napi erőnléti szintje. Bár az első néhány szakaszon (jellemzően az út első 10%-án) az ingadozó fittségi faktor értékek miatt még pontatlan az eljárás, a továbbiakban - mint azt látni fogjuk - igen jó becslést adhatunk a menetidőre. A kísérlet során megpróbáltuk ezt a fittségi faktort nem csupán "globálisan" meghatározni egy adott túrázó esetén, de akár meredekségi intervallumokra külön-külön. Gyakorlati tapasztalatunk szerint enélkül ugyanis figyelmen kívül hagyjuk, ha valaki sík terepen kiválóan teljesít ugyan, de az emelkedőkre rosszul reagál. Mivel azonban így több faktort is becsülnünk kell a túra során, ezért mire azok értékei "stabilizálódnak", már jellemzően igen sok szakaszt megtett a túrázó, így összességében ezzel a kiterjesztéssel rosszabb eredményeket értünk el, mintha csak egy fittségi faktort becsülnénk.

2. Az átlagsebesség alapú menetidőbecslés

Az eljárás a túra első 20%-ában az illesztett sebességgörbe alapján becsli a sebességet a teljes útra, miközben minden szakasz sebesség értékeit elraktározza. Legyen az i-edik, már megtett szakasz megfigyelt sebessége v_i , ekkor az n-edik szakasz (mely már túl van a túra első 1/5-én) sebességét az alábbiak szerint becsüljük:

$$v_n^* = \frac{\sum_{i=1}^n v_{i-1}}{n-1}$$

tehát egyszerűen vesszük az előző n-1 szakaszon megfigyelt sebességek átlagát. Jellemzően a túra első 1/5-ében ez az eljárás még nem ad jó sebességbecslést, ezért kell ott helyettesítenünk az illesztett sebességgörbe által adott becsléssel. A következő alfejezetben összegezzük a becsülő eljárásaink jóságának vizsgálatait.

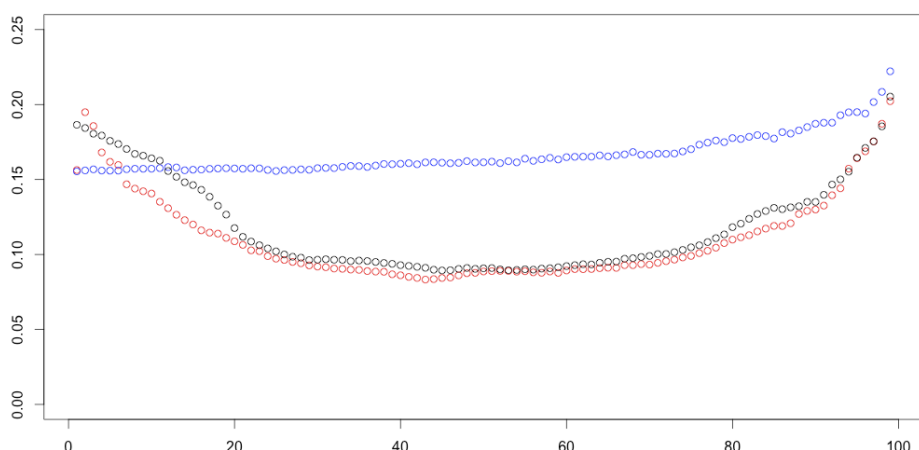
Az eredmények kiértékelése

Becslésünk pontosságát mérendő, szeretnénk azt a Tobler-görbe alapján kalkulált becslésekkel összevetni. Összehasonlítási mértékként mi is (akárcsak Pittman et al. [7]) az átlagos abszolút relatív hiba (mean absolute relative error, röviden MARE) értékét használjuk, mert egyformán bünteti az alul- és felülbecslést is. A teljes utat 100 részre bontjuk, és p -vel jelöljük, hogy az út hány százalékánál tartunk. Az i -edik útra a p -edik szakaszhoz tartozó, mért adatokon alapuló hátralévő időt jelölje r_{ip} míg az általunk becsült hátralévő időt r_{ip}^* .

$$MARE(p) = \frac{1}{n} \sum_{i=1}^n \left| \frac{r_{ip} - r_{ip}^*}{r_{ip}} \right|$$

A számításnál a 2400 túrából álló adatbázisunkat tanuló- és teszt adathalmazra (75-25%) bontottuk véletlen módon. A tanuló adathalmaz alapján végeztük a 2.5.3-as szakaszban bemutatott görbe illesztését a sebesség átlagokra (a meredekség függvényében), majd az így kapott görbét alkalmaztuk a teszt adathalmazon a már ismertett két módszer szerinti menetidőbecslő eljárások során. Ezt az eljárást 10-szer alkalmaztuk egymás után, és a fenti képlet alapján kalkulált MARE értékeket. A MARE értékek azt mutatják, hogy az illesztett sebességgörbén alapuló eljárás teljesít a legjobban (11,58%), míg azt nem sokkal lemaradva követi az átlagsebességre épülő becslésünk (12,94%). Mindkét eljárás által becsült eredmények szignifikánsan jobbak Tobler eredményeinél (17,36%), a Naismith-szabály által prognosztizált menetidőknél, sőt a Pitmanék által javasolt módszer eredményeinél is átlagosan nagyjából 5 százalékponttal jobb becslést ad, bár utóbbit sajnos nem tudtuk összevetni a saját eredményeinkkel azonos adatbázison végzett tesztekkel. A teszt túranaplókon végzett menetidőbecslések MARE értékeit mindhárom vizsgált eljárásra a 6. ábrán foglaltuk össze (a piros a meredekségen, a fekete az átlagsebességen alapuló eljárást jelöli, míg késsel tüntettük fel a Tobler-görbe alapján becsült menetidők MARE értékeit)

6.ábra: A három eljárás MARE értékei (a megtett út %-ának függvényében)



2.2. Turisztikai ajánlórendszer

Jelen szakaszt az ajánlórendszerekre vonatkozó saját definícióval kezdjük

Ajánló rendszer: olyan információsűrű rendszer, mely egy adott döntési helyzetben a lehetséges opciók halmazának szűkítésével, illetve az elemeinek adott kontextusban történő rangsorolásával támogatja a felhasználót. A rangsorolás történhet a felhasználó explicit vagy implicit módon kifejezett preferenciái alapján, illetve a hozzá hasonló preferenciákkal bíró felhasználók korábbi viselkedésének figyelembe vételével.

Egy potenciálisan sikeres turisztikai alkalmazás egyik előfeltétele, hogy a lehető legnagyobb pontossággal tudja a felhasználók számára releváns, érdeklődésükre számot tartó célpontokat kínálni. Ez azonban koránt sem olyan egyszerű, amennyiben nem áll rendelkezésünkre kellő információ az adott felhasználó ízlésvilágáról. A preferenciák feltérképezésének egyik fő kihívása, hogy ezt a felhasználók idejének lehetőleg minimális igénybevételével oldjuk meg, hiszen célunk éppen az lenne, hogy időt és energiát takarítsunk meg számukra a tervezés során. A legtöbb turisztikai ajánlórendszer általában a teljes felhasználói csoport értékelése alapján kialakított ranglistát tekinti irányadónak minden felhasználója számára, és nem fordít figyelmet az egyéni preferenciákra. Mi ezzel ellentétben személyre szabott ajánlásokat kívánunk tenni, ennek érdekében az általunk javasolt 17 turisztikai kategóriát a 1. táblázatban foglaljuk össze:

Kiemelkedő látványosság	Templom/vallási témájú hely	Múzeum/művészet
Történelem/kultúra	Építészet/épület/homlokzat	Történelmi helyszín/emlékmű
Utcák/terek	Kilátópont	Természet/park
Egyetem/tudomány/technológia	Család/gyermek program	Fürdő/sport/rekreáció
Piac/helyi ételek	Kávézó/étterem	Színház/mozi/szórakozás
Éjszakai élet/zene/bár	Vásárlás/divat	

A fenti kategóriák jó tematikus lefedését adják a helyszíneknek, és kombinációjukkal az összes látványosság jól leírható eddigi tapasztalataink alapján

A felhasznált adatok

Empirikus vizsgálatunk során 3 városra állítottunk össze látványosságokat tartalmazó, részletes adatbázist, mely az ajánlórendszer tartalom alapú modulját alapozza meg.

Az ajánlórendszer teszteléséhez 3 város (Budapest, London és Párizs) összesen 500 helyszínéből álló adathalmazt hoztunk létre, mely az alábbi változókat tartalmazza:

- **POI_Name**: a helyszín neve
- **Category_i**: a helyszín kategóriája (minimum egy, legfeljebb 3 ilyen kategória lehetséges), melyek a 3.8.1-es alfejezetben meghatározott 17 kategóriából kerültek kiválasztásra, (például a Nagytétényi kastély esetén ezek: Építészet/épület/homlokzat, Múzeum/művészet, Történelem/kultúra)
- **Relevance_i**: az egyes kategóriákhoz tartozó relevancia értékek, vagyis, hogy mennyire írja le jól az adott kategória a helyszínt, értéke 0-3 közötti egész szám, (a Nagytétényi kastély példájánál maradva ezek a relevancia értékek rendre: 3, 3, 2)
- **Importance**: a helyszín fontossága, azaz, hogy mennyire számít ismertnek, kiemeltnek egy adott látványosság, (ez mérhető például a látványosságot leíró wikipedia oldal hosszában, vagy hogy mennyi egyedi találatot ad a google keresője, illetve később a látogatók számából is következtethetünk rá), értéke 1, 2 vagy 3 lehet (3 jelöli a legfontosabb helyeket)
- **City**: a város, ahol a látványosság található
- **WikiLink**: a látványosság wikipedia oldala, mely a kísérlet során tájékoztatást ad az alanyoknak a látványosságról, ha azt esetleg nem ismerik.

Az empirikus vizsgálat ismertetése

Célunk egy olyan ajánlórendszer megtervezése, mely alkalmas a felhasználók széles körét kiszolgálni. Ez szerves részét képezi majd annak a célfüggvénynek, melyet a 4. fejezetben igyekszünk maximalizálni a felhasználó igényeit kielégítő, egyéni útvonaltervezéssel, hiszen a látványosságok értékelései ebben a szakaszban kerülnek meghatározásra. A felhasználók helyszínekre vonatkozó értékeléseit minél pontosabban leíró kalkulációs eljárások vizsgálatát ebben a szakaszban mutatjuk be.

Az általunk vizsgált eljárások alapja az imént ismertetett adatbázis, mely azon a feltételezésen alapszik, hogy a turisztikai látványosságok jellemezhetőek különböző faktorok segítségével, ahogyan például a zene tulajdonságainak feltárása történt a Music Genome Project során [8]. Ezt bővítjük a felhasználók ezen faktorokra vonatkozó preferenciáinak begyűjtésével

kérdezőes módszerrel. Így a vizsgálat alá vont eljárásunk egy hibrid ajánlórendszer, mely az alábbi egységekből áll:

- Tudás alapú modul [10]: adatgyűjtési eljárás során a felhasználóktól tudás alapú kérdező eljárással kapjuk meg a megállapított 17 faktorra vonatkozó értékelésüket, (illetve a későbbi szakaszban egyéb paramétereket is, például a napi költségkeret összegét, és az időkorlátjukat is, mely az útvonaltervezéshez szükséges majd).
- Tartalom alapú modul [11]: a látványosságok faktorok kombinációjaként történő leírása (ahol a súlyok a faktorokra vonatkozó relevancia értékek) a tartalom alapú szűrési technika előfeltétele. Ez lehetőséget nyújt olyan ajánlások készítésére, mely során a felhasználónak korábbi visszajelzései alapján olyan helyszíneket ajánlunk, melyhez hasonlóakat korábban már pozitívan értékelt. Ekkor a 17 faktort reprezentáló 17 elemű vektorba rendezzük sorban a hozzájuk tartozó relevancia értékeket, és vektor-cosinus eljárással [9] keresünk a korábban a felhasználó által pozitívan értékelt látványosságokat reprezentáló vektorokhoz hasonlóakat, (vagyis olyanokat, melyek kis szöveget zárnak be azokkal).

Ez a technika a korábban ismertetett hibrid eljárások osztályozásában a *tulajdonság kiterjesztő eljárások* (Feature-augmenting) közé tartozik [11], hiszen minden ajánlható elem értékeléseit egy másik technikával minden látványosságra új tulajdonságvektort alkothat, melyet az alapeljárás során (a kibővített információval együtt) felhasználunk. Nagy előnye az eljárásnak, hogy a tudás- és tartalom alapú megközelítések ötvözésének köszönhetően minimális információval el tud indulni a rendszer, gyakorlatilag csak a felhasználó 17 faktorra vonatkozó értékelésére és a célvárosra van szükség kezdeti információként. Cserébe az induló adatbázis előkészítése időigényes, hiszen a látványosságokat leíró faktorokat és azok kezdeti relevancia értékeit meg kell adni. Ezek a későbbiekben a felhasználók értékelései alapján finomíthatóak, ahogy egyre több információhoz jutunk a velük történő interakciók során. Ennek az ajánlórendszernek sikeressége - az adatbázis mellett - azon a függvényen múlik, amellyel a felhasználó által az egyes faktorokra adott értékelések alapján a látványosságokat pontozzuk. A kutatás jelen szakaszában az alábbi 4 kalkulációs eljárás által adott ajánlásokat kell értékelniük a résztvevőknek:

- *KPII*: A felhasználó által az egyes faktorokra adott értékelések szorzata az adott faktor helyszínre vonatkozó relevanciájával összegezve az összes faktorra, majd ezt szorozzuk a helyszín fontosságával, (*imp* a helyszín fontossága, e_i az *i*-edik faktorra adott értékelés, és r_i az *i*-edik faktor relevanciája). Továbbá minden olyan látványosság értékelését 50%-kal

csökkentjük, aminek legalább 1 faktorából, melyhez legalább 2-es relevancia érték társul, már legalább 5 szerepel a kiválasztott látványosságok listáján. Ezzel azt kívánjuk elkerülni, hogy túlzottan egysíkú ajánlásokat tegyünk, és idővel büntetjük a hasonló elemeket.

$$imp \sum_{i=1}^{17} e_i r_i$$

- *KPI2*: Az első eljárás során kapott pontot megduplázzuk a 2-es fontosság érték esetén, ezzel próbáljuk előnyben részesíteni a kevésbé ismert helyszíneket, (legyen tehát $h=1$, ha $imp=2$, és 0 különben). Ezt elosztjuk a 0-tól különböző értékelést kapott faktorok számával, amit jelöljön k . Ezzel igyekszünk azoknak a helyszíneknek a hátrányát lefaragni, melyek bár bizonyos szempontból relevánsak és izgalmasak, de nem tartozik hozzájuk több faktor. Ezt kiegészítjük még azzal, hogy a helyszín pontszámának értékét felezzük, amennyiben van olyan 3-as relevanciájú faktora, mely faktorhoz a felhasználó 0 értékelést rendelt.

$$\frac{imp \sum_{i=1}^{17} e_i r_i (1 + h)}{k}$$

- *KPI3*: A *KPI2* eljáráshoz képest annyit módosítunk, hogy növeljük a kifejezés értékét annyszorosával, ahány 9-es értéket találunk a relevancia-faktor értékelés szorzatok között, ezek számát jelölje l :

$$\frac{imp \sum_{i=1}^{17} e_i r_i (1 + h) (1 + l)}{k}$$

- *KPI4*: Az utolsó eljárás első 7 eleme a *KPI1* eljárás listájának első 7 eleme, míg további 8 elemét a *KPI2* eljárás szerint kalkulált listából vesszük sorrendben úgy, hogy duplikáció ne forduljon elő.

A fenti 4 eljárás által adott ajánlások vizsgálata érdekében létrehoztunk egy weboldalt (www.travelschedule.org), ahol a korábban ismertetett három város látványosságait értékelhetik. Összegezve a teljes populációra nézve Budapest esetén az eljárások sorrendje (értékelés és rangsor alapján számolva egyaránt): 2-3-4-1, míg küldöldön ez éppen ellentétes: 1-4-3-2. A faktorokra adott értékelések alapján azonban tovább finomíthatjuk az eredményeinket, és lehetőségünk nyílik megérteni az egyes turisták motivációit.

A faktorok közötti összefüggések megértése érdekében képezzük azok korrelációs mátrixát, (zölddel jelöltük a 0,4 fölötti korrelációs együtthatókat, és sárgával a 0,3-0,4 közöttieket). Az egymással csoporton belül korreláló faktorok alapján 6 turista típust tudtunk azonosítani. Az egyes csoportok tagjai által az eljárásokra adott értékelések viszont jól elválnak egymástól. A 2. táblázatban foglaljuk össze, milyen sorrendet állíthatunk fel az eljárások között az egyes csoportok esetében. Jól látható, hogy a külföldi értékelések körében minden esetben az első eljárás volt a legnépszerűbb, míg a budapesti értékelések esetén 3-3 csoport értékelte legjobbnak a 2. és 4. eljárást. A csoportok között előforduló átfedések miatt azonban esetenként nehéz eldönteni, mely eljárással adható a legjobb ajánlás. Szükség mintánkon végzett szimulációink alapján az alábbi mechanizmus bizonyult a legsikeresebbnek: Külföldi helyszín esetén válasszuk az 1. eljárást. Budapest esetén ha valaki összes leadott értékelése meghaladja a 2,2 pontot, akkor adjuk ismét az 1. ajánlást. Ha művészet kedvelőként klasszifikáltuk (akár egyebek mellett), adjuk a 2. ajánlást. Ha nem művészet kedvelő, de fiatalként klasszifikáltuk, adjuk a 4. ajánlást. Ha a fentiek közül egyik sem, adjuk a 2. ajánlást. Ezzel a mechanizmussal a 88 esetből 76-ban adtuk a legjobbra értékelt ajánlást, és csak 2 esetben nem a 2. legjobbat. Ha a priori minden esetben a móduszt ajánljuk (tehát Budapest esetén a 2. eljárást, míg külföld esetén az 1. eljárást), akkor 88 esetből csak 36 esetben adtuk volna a felhasználó számára legjobb ajánlást.

2. Táblázat: Azonosított turista típusok eljárásokra vonatkozó értékeléseinek sorrendje							
Turisztikai faktorok	Eljárások	Kultúra kedvelő	Természet kedvelő	Családias	Fiatal	Mondén	Gurmé
Bp	KPI1	3	4	2	4	4	4
	KPI2	1	1	1	3	3	2
	KPI3	2	2	4	2	2	3
	KPI4	4	3	3	1	1	1
London - Párizs	KPI1	1	1	1	1	1	1
	KPI2	4	4	4	3	4	3
	KPI3	3	3	3	4	2	2
	KPI4	2	2	2	2	3	4

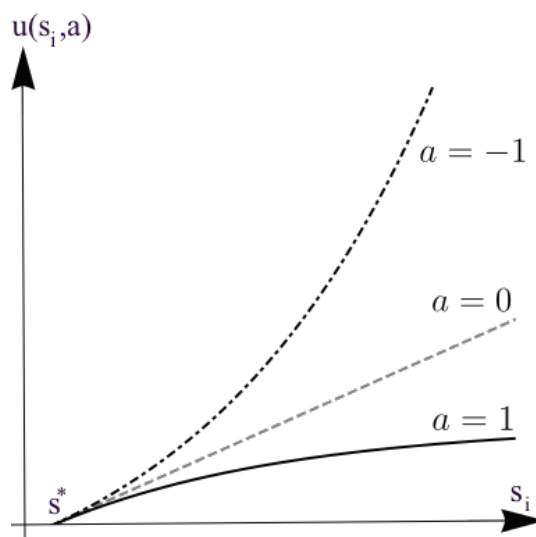
2.3. Útvonaltervezés

Az alábbiakban bemutatásra kerül a dolgozat magját képező útvonaltervező algoritmus, melyhez felhasználjuk a korábbi fejezetek eredményeit is, így a tervezés alapjául szolgáló gráf élköltségeit az útvonaltervező algoritmus segítségével határozzuk meg, míg a helyszíneket jelképező csúcsok értékelései a 3. fejezetben leírt ajánlórendszer segítségével kalkulálhatóak. Elsőként bemutatjuk a tervezéshez felhasznált adatokat, majd a probléma megfogalmazása után egy heurisztikus eljárást adunk annak megoldására.

A feladat nem más, mint az utazóügynök problémában is meghatározott csúcsoknál gyűjthető profitok összegének maximalizálása. Ez a megközelítés még a TSP megfogalmazása óta része az útvonaltervező algoritmusoknak, ahol pénzben mérhető profitról lévén szó, összeadható volt, és reális elvárás, hogy az összprofitot maximalizáljuk. Helyszínekre adott értékelések esetén azonban ez már nem igaz. Javasolom tehát, hogy ne “pontgyűjtő akcióként” kezeljük a feladatot, és ennek érdekében egy olyan célfüggvényt alakítsunk ki, mely igyekszik garantálni a felhasználót leginkább kielégítő útvonal megtervezését (7. ábra):

$$u(s_i, a) = \begin{cases} \frac{1 - e^{-a(s_i - s^*)}}{a} & | a \neq 0 \\ s_i - s^* & | a = 0 \end{cases}$$

7. ábra: Hasznossági függvény



Az a elméletben $(-\infty; \infty)$ intervallumom bármilyen értéket felvehet, gyakorlati megfontolások alapján $[-2; 2]$ intervallumban vizsgáljuk majd az útvonaltervre gyakorolt hatását. Mivel

$u(s^*, a) = 0$, így tehát azok a helyszínek, melyek értékelése küszöbértéken van, 0 profitot hoznak, és csak az ennél jobb értékelés helyszínek jelentenek pozitív profitot, melyek növelik a célfüggvény értékét. Ez függvényforma természetesen csak javaslat, ám a kutatás jelen fázisának eredményei alapján reményt keltő annak alkalmazása.

Ezeket figyelembe véve a célfüggvényünk, mely alapján az újabb pontokat vesszük fel az útvonalba:

$$C(\alpha, \beta, a, R) = \left(\frac{\sum_{p=1}^P \sum_{i=1}^N \theta_{ip} v_i}{\sum_{i=1}^{N-1} \sum_{j=2}^N \tau_{ijp} t_{ij}^\beta} \right)^\alpha \times \left(\sum_{p=1}^P \sum_{i=1}^N \theta_{ip} u(s_i, a) \right)^{1-\alpha}$$

ahol P a rendelkezésre álló napok száma, N a gráf csúcsainak száma, s_i , v_i , t_i rendre a következő pont értékelése, látogatási ideje és az odaút menetideje, s^* a felhasználó értékeléseinek azon küszöbértéke, ami alatt nem kerülhet be látványosság a potenciálisan látogatható helyszínek közé, α a célfüggvényben szereplő két szempont (a hasznosság és az egységnyi menetidőre eső látogatási idő) súlyozására szolgál, β a “lustasági” paraméter. A τ_{ijp} értéke legyen 1, ha a p -edik útnál az i -edik csúc után a j -edik következik az úton, és 0 különben. Legyen θ_{ip} értéke 1, ha a p -edik úton az i -edik csúcsot meglátogatják, és 0 különben. A P napra tervezett útvonalak összességét R jelöli. Az a paraméter hivatott tükrözni, mennyivel értékelt többre a felhasználó egy s -re értékelt látványosságot egy $(s-1)$ -re értékkelthez képest.

A választott célfüggvény forma tehát alapvetően két törekvést szolgál: egyrészt az egységnyi megtett útra jutó fajlagos látogatási időt igyekszik növelni, másrészt a legnagyobb hasznossággal bíró csúcsok meglátogatását szorgalmazza. Az α paraméterrel ezek súlyát szabályozhatjuk. Fontos látni, hogy a célfüggvény a mások által széles körben használt profitösszeg-maximalizálás egy kiterjesztése, hiszen $\alpha = a = 0$ választással éppen ezt kapjuk.

A feladat formalizálása

Legyen adott egy $G(V, E)$ gráf, amelynek minden c_i csúcsához egy s_i nemnegatív értékelés van rendelve, mely a turista számára $u(s_i, a)$ hasznossággal bír, ha meglátogatja a c_i csúcsot. A c_i és c_j csúcsok közötti e_{ij} élhez t_{ij} élköltséget rendelünk, ami a turista számára a távolság megtételéhez szükséges idő. Az c_i csúcs meglátogatása v_i időt vesz igénybe (látogatási idő). 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 θ_{ip} értéke 1, ha a p -edik úton az i -edik csúcsot meglátogatják, és 0 különben. A turistának P napja van a látványosságok megtekintésére, és naponta T_{max} perc ideje. Az i -edik látványosság megtekintése b_i költséggel jár (belépődíj), melyet a napi B költségkeretéből fedezhet. Fontos, hogy a költségvetési korlát tetszés szerint átcsoportosítható a napok között, így összességében a P napra BP költségkerettel rendelkezik. Ez nem vonatkozik az időkorlátra, mely minden napon betartandó. Minden nap elején a szállodából indul, és a nap végén oda érkezik vissza. A modellben ezt az általánosság jegyében külön kezeljük (az 1-es és N -nel jelölt csúcs), de ezek megegyezhetnek egymással. A feladat, hogy P nap alatt olyan P darab utat bejárni a $G(V,E)$ gráfon, hogy maximalizáljuk a célfüggvény értékét, miközben betartjuk az idő- és költségkorlátokat, és minden csúcs legfeljebb egyszer látogatható meg. Ekkor a feladat megfogalmazható a következőképpen:

$$\max_{\tau_{ijp}} \left(\frac{\sum_{p=1}^P \sum_{i=1}^N \theta_{ip} v_i}{\sum_{i=1}^{N-1} \sum_{j=2}^N \tau_{ijp} t_{ij}^\beta} \right)^\alpha \times \left(\sum_{p=1}^P \sum_{i=1}^N \theta_{ip} u(s_i, a) \right)^{1-\alpha}$$

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

$$\sum_{p=1}^P \theta_{kp} \leq 1; \forall k = 2, \dots, N-1$$

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

$$\sum_{i=1}^{N-1} \sum_{j=2}^N \tau_{ijp} t_{ij} + \sum_{i=1}^N \theta_{ip} v_i \leq T_{max}; \forall p = 1, \dots, P$$

$$\sum_{p=1}^P \sum_{i=1}^N \theta_{ip} b_i \leq PB$$

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

$$2 \leq h_{ip} \leq N; \forall i = 2, \dots, N; \forall p = 1, \dots, P$$

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

Az egyes kifejezések jelentése a következő:

1. A maximalizálandó célfüggvény
2. Minden út az I -es csúcsnál kezdődik, és az N -ediknél ér véget, (ezek a korábbiak alapján megegyezhetnek).
3. Minden csúcsot csak legfeljebb egyszer látogatunk meg.
4. Minden út egyenként összefüggő.
5. Betartjuk az időkorlátot: a napi látogatási- és menetidők összege nem lehet több, mint T_{max} .
6. Betartjuk a költségkorlátot: a belépődíjak összege a P napra együttesen nem lehet több BP -nél.
7. és 8. együtt garantálja, hogy ne legyenek körök az útban, Miller–Tucker–Zemlin javaslata alapján [14].
9. A τ_{ijp} és θ_{ip} értékészlete 0 vagy 1.

A kitűzött feladatra adott heurisztikus megoldásunkat a következő alfejezetben ismertetjük.

Az útvonaltervezés

Először két olyan eljárást ismertetünk, melyet több ponton használunk majd az algoritmus során:

Lexikografikus rendezés: ennek során mindig egy csúcsokból álló halmazt rendezünk egy másik csúcsokból álló halmaz és az erőforrás keretek (pénz és idő) szűkössége alapján (mely meghatározásának menetét később pontosan ismertetjük). Szükségünk van továbbá arra az s^* küszöbértékre, melynél alacsonyabb értékelésű csúcsokat törölni fogunk a releváns pontok halmazából (lásd, az algoritmus első lépése). Formálisan tehát $L(C_1, C_2, sc, s^*)$, ahol C_1 a csúcsok azon halmaza, melyet rendezni szeretnénk, C_2 amely halmazhoz rendezzük, sc az erőforrások szűkösségének mértékét állítja sorrendbe (idő vagy pénz), és s^* az értékelések küszöbértéke. Képezzük az alábbi értékeket:

$$\frac{\frac{u(s_i, a)}{u(s^*+1, a)}}{d^*(c_i, C_2) + v_i} \quad \frac{\frac{u(s_i, a)}{u(s^*+1, a)}}{b_i}$$

ahol $d^*(c_i, C_2)$ a c_i csúcs és a C_2 halmaz közötti átlagos távolságot jelöli. A számláló tehát azt fejezi ki, hogy az adott s_i értékelésű pont hány s^*+1 értékelésű pont hasznosságával

egyenértékű (a releváns pontok között ugyanis s^*+1 értékelésű a minimális, hiszen az s^* értékelésűeket és az annál kisebbeket töröljük a gráfból). Ezt osztjuk a nevezőben a pont felvételének várható költségével, ami az odaút menetideje plusz a látogatási idő. A pénzben kifejezett költségek rendezésekor ugyanezen az elven a nevezőben a belépődíj szerepel. A következő lépésben rendezzük a C_I halmaz csúcsait, először aszerint, amelyik korlát szűkösebb. Ha ez például az idő, akkor a fenti hasznosság/időköltség mutató alapján rendezzük csökkenő sorrendbe, majd az így kapott listát nagyjából 6 egyenlő részre osztjuk kvantilisek segítségével (csak az utolsó csoport elemszáma különbözhet a többitől). A második korláthoz tartozó mutató alapján is sorba rendezzük a csúcsokat a 6 csoporton belül. A csoportosításra azért van szükség, mert az első mutatószám értékei alapján már egyértelműen sorba rendezhetjük általában a csúcsokat, így azok kis eltérése esetén sem lenne lehetőségünk a lexikografikus rendezésnél a második mutató alapján felülvizsgálni a sorrendet.

Outlier számítás: Az outlier kereső eljárásunk $O(H, cr)$ egy adott H Hamilton-kör outlier értékeit adja meg egy cr kritikus időérték mellett. Meghatározzuk H minden csúcsára a ki- és bemenő élek összidejét, majd azok átlagát és szórását. Azon i elemeket tartjuk outliernek, melyek esetén

$$t_{i,be} + t_{i,ki} > t_H^* + cr\sigma_H$$

ahol t_H^* az átlagos be- és kimenő élköltség és σ_H a szórás. A cr értéke az algoritmus egyes lépéseinél változhat. Ezt külön jelezzük majd.

Heurisztikus algoritmusunk az alábbi lépésekből áll:

1. A probléma egyszerűsítése: töröljük a gráf minden csúcsát (az azokba bemenő és azokból kimenő élekkel együtt), melynek értékelése kisebb vagy egyenlő s^* küszöbértéknél, melyet a gyakorlatban választhatunk úgy, hogy a megmaradó csúcsok összes látogatási ideje ne haladja meg a PT_{max} rendelkezésre álló összidőkeret kétszeresét. Gyakorlati tapasztalatunk alapján az ennél több pont szerepeltetése nem javít az optimumon, ellenben az algoritmus számítási igényét fölöslegesen növeli. Ennél általában konzervatívabb megoldás, ha s^* értékét úgy választjuk, hogy megegyezzen az értékelések átlagával, hiszen ha $s_i < s^*$, akkor $u(s_i, a) < 0$, tehát nem javíthat a célfüggvény értéken. (Az általunk vizsgált esetekben mindig volt annyi átlagon felüli értékelésű csúcs, hogy azok összes látogatási ideje meghaladja a PT_{max}

rendelkezésre álló összidőkeret kétszeresét.) Az így kapott halmazt a továbbiakban a releváns csúcsok halmazának nevezzük.

2. Fix csúcsok: Rögzítsük a kötelezően meglátogatandó csúcsokat. Ezek az eljárás során soha nem kerülhetnek az útvonalból törlésre. A korábbi megegyezés alapján azokat tekintjük kötelezőnek, melyekre vonatkozóan a turista értékelése maximális volt.

3. Csoportosítás: A releváns csúcsok alapján megbecsüljük, melyik erőforrás korlátunk szűkösebb. Ennek érdekében összevetjük az alábbi két hányadost:

- a releváns csúcsok látogatási idejéhez hozzáadjuk a releváns csúcsok közötti átlagos távolságot, és ezt az összeget szorozzuk a releváns csúcsok számával, majd elosztjuk a rendelkezésre álló PT_{max} időkerettel,
- a releváns csúcsok látogatási költségének összegét osztjuk a BP költségvetési kerettel.

Amelyik érték nagyobb, azt tekintjük szűkösebb korlátnak, és a lexikografikus rendezések alkalmával a pontszámok után rögtön azt a korlátot vesszük előre. Ha tehát a szűkös korlát az idő, akkor a rendezésnél az alábbi sorrendet vesszük figyelembe a változók körében: értékelés, idő, pénz. A maximális pontszámú (tehát kötelező) csúcsok mellé a maradék releváns csúcsra lexikografikus rendezés után választjuk az első $5P$ darab csúcsot. Ez az egyetlen olyan lépés, ahol a fent ismertetett lexikografikus elrendezéstől eltértünk annyiban, hogy első rendező elvként a pontszámot használtuk. A releváns pontok halmazának outlier csúcsait rendhagyó módon egy a teljes halmazra meghatározott legrövidebb Hamilton-kör meghatározásával kezdjük (melynek kezdő- és végpontja a szálloda). Ebből $cr = 1$ értékválasztás mellett használjuk az $O(H, cr)$ függvényt az outlierok kiszűrésére (természetesen csak a nem maximális értékelésű csúcsok lehetnek outlierok).

4. Napi utak építése: A megmaradó csúcsokat P darab (napok száma) klaszterre bontjuk Hartigan-Wong klaszterező eljárással [15], melyet relatív hatékonysága miatt választottunk. Minden klaszterre kiszámoljuk a szállodával alkotott legrövidebb Hamilton-kört a TSP megoldására adott algoritmussal. Ennek háttérében az a megfontolás áll, hogy amennyiben csak egy fix csúcsokból álló halmazon szeretnénk a célfüggvényünket maximalizálni, az ekvivalens a csúcsokat összekötő élek élköltségeinek β -adik hatványaival vett gráfon történő legrövidebb Hamilton-kör meghatározásával, hiszen a csúcsok változatlansága miatt mind a

hasznosságok, mind a látogatási idők értéke állandó az adott halmazra. Ezt kihasználva az R szoftverben beépített TSP optimalizáló (Repetitive Nearest Neighbor Algorithm) eljárást alkalmaztuk [16]. Ezt a klaszterező eljárást 10-szer ismétljük meg, hiszen a klaszterezés gyakran vezet különböző eredményre. Az 10 eredmény közül azt választjuk, ahol P darab Hamilton-körre számolt célfüggvény értékünk maximális.

5. Feltöltés: Az előző lépésben P darab utat kaptunk, mely a szállodánál kezdődik és ott ér véget. Amennyiben még nem értük el a napi idő- és pénzkeret 1,2-szeresét, akkor az $L(C_r, C_i, sc, s^*)$ eljárással rendezzük az i -edik napra a releváns csúcsok halmazát, melyeket még egy útba sem illesztettünk be (jelölje ezt C_r). Itt fontos megemlíteni két elvet:

- Azokat a csúcsokat, melyeket egy adott napra beillesztünk, automatikusan kivesszük a még megmaradt releváns csúcsok C_r halmazából.
- Ha egy csúcsot kivesszünk egy naphól, azt automatikusan visszarakjuk a C_r halmazba.

Így minden napra rendeztük C_r elemeit, és a lista elejéről kezdve elkezdjük feltölteni a csúcsokkal a napokat, amíg el nem érjük az idő- és pénzkorlát 1,2-szeresét. Amennyiben egy csúcs két nap szerinti rendezésben is bekerülne az útba, oda helyezzük, ahol magasabb marginális célfüggvény javulást eredményez. Az i -edik napra való felvétel kritériuma, hogy az így keletkező Hamilton-körben az adott pont ne legyen outlier, ahol az $O(H, cr)$ függvényt $cr = 1,5$ mellett értékeljük ki.

6. Csere: Minden nap csúcsaira meghatározzuk a többi nap pontjaitól vett 3 legkisebb érték átlagát, ezt a csúcs saját napjára is kiszámítjuk (ahol a másodiktól a negyedik legkisebb értékig vesszük az átlagot, hiszen a legkisebb érték, az önmagával vett távolság, ami 0). Ezután minden csúcsot arra a napra helyezzük át, hol ez az érték minimális. Ezt az iterációt 10-szer ismétljük meg egymás után. **7. Levágás:** Ha van olyan nap, ahol meghaladtuk a napi időkeretet több mint 5%-kal (ennyit engedélyezünk legfeljebb), akkor az $L(C_i, C_i, sc, s^*)$ alapján (vagyis saját magával) rendezve a naphoz tartozó csúcsok halmazát az utolsó elemeket addig távolítjuk el a napi útból, míg az időkihasználása a keret 105%-ánál nem lesz kevesebb. A pénzkorlát túllépése esetén az(oka)t a ponto(ka)t távolítjuk el, ahol az egységnyi pénzköltségre eső marginális célfüggvény növekedés minimális

8. Feltöltés: Amennyiben van olyan nap, ahol még van szabad időkapacitás, a C_r elemeit $L(C_r, C_i, sc, s^*)$ eljárással rendezzük az i -edik napra, és az első elemtől kezdve elkezdjük a

napot feltölteni, míg az időkorlátot és a P napra szánt költségvetési korlátot át nem lépjük. Itt ismét a felvétel kritériumaként az $O(H, 1,5)$ függvényt alkalmazzuk, mint korábban.

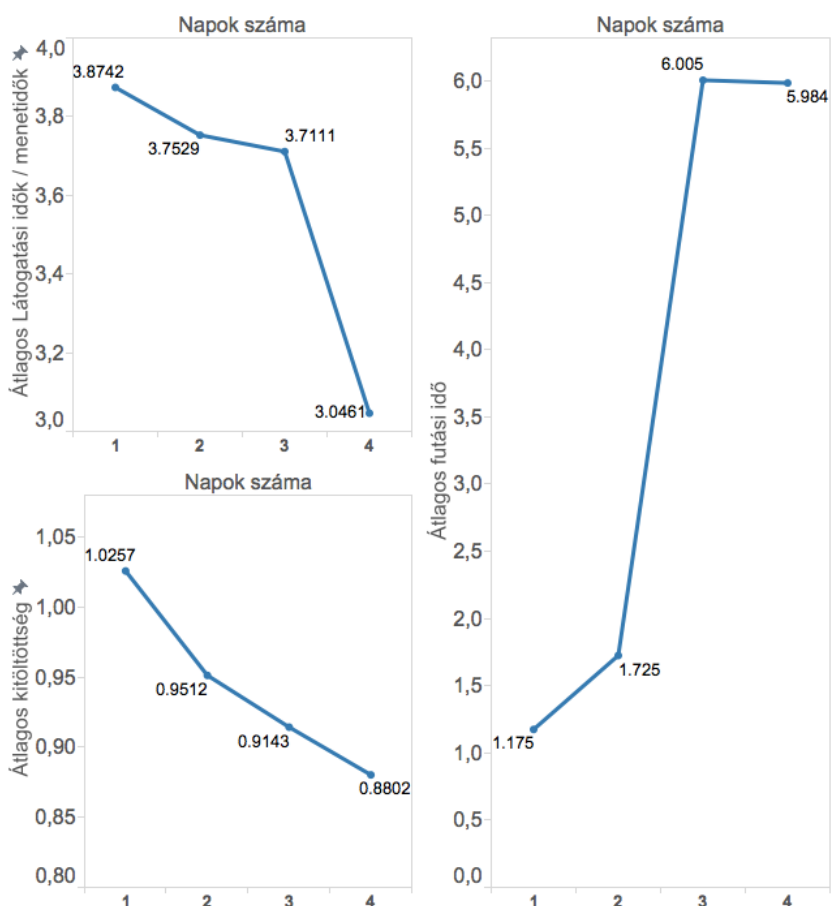
Az eredmények kiértékelése

A fenti algoritmust a 150 budapesti turisztikai látványosságot tartalmazó adathalmazon tesztelve az alábbi megállapításokat tehetjük:

- Pozitív a értékek (konkáv hasznossági görbe) jellemzően nagy kitérőket eredményeznek
- Amennyiben ezek alacsony α és β értékekkel párosulnak, úgy az utak “szétesőek”, tehát viszonylag kevés pontot, és nagy élkötségeket tartalmaznak.
- Viszonylag kis α értékek (0,5 alatt) csak magas ($>1,5$) β és alacsony a ($< - 0,5$) értékek mellett ad “jó” megoldást.
- Általában elmondható, hogy $a < - 0,5$; $\alpha > 0,5$ és $\beta > 1,5$ esetén kaptunk jó megoldásokat.
- Az $\alpha = 0,75$; $a = -1$ és $\beta = 2$ választása mellett adja összességében 1-2-3 és 4 napos túrákra a legjobb megoldást (hosszabb túrákat nem vizsgáltunk), ahol a napok feltöltöttsége is igen jó, és a teljes túrákra kalkulált látogatási idő - menetidő hányados is kiemelkedően magas. Összehasonlításként az $\alpha = a = 0$ esettel, mely a szakirodalomban széles körben elterjedt profitösszeg-maximalizálást jelenti elmondhatjuk, hogy a profitösszeg maximalizálás jóval alacsonyabb látogatási idő - menetidő arányt eredményez (átlagosan 1,43 szemben az általunk kiemelt eset átlagosan 3,8-es eredményével), a napok kitöltöttsége mindkét esetben átlagosan 95% körüli, és a futási idő érthető módon átlagosan nagyjából 1 másodperccel hosszabb az általunk választott paraméterek esetében.
- Az algoritmus 3 napos útvonal megtervezését 100-szor futtatva átlagosan 3,73 másodperc alatt végezte el. Sajnos ezt nem tudjuk összevetni Vansteenwegen et al. [12] vagy Gavalas et al. [140] eredményeivel, hiszen merőben más feladatot oldottunk meg, sőt azok optimális megoldása számolható LP feladatként, míg célfüggvényünk miatt a miénk egy NLP feladat. Az összevetés kedvéért egy másik mobil applikációhoz tervezett heurisztikus eljárást említve Sylejmani és Dika 2011-es cikkében [2] 40 látványosságra tervezett 3 napos túrájuk számítási ideje átlagosan 81,7 másodperc volt Tabu search algoritmussal.
- A futási időről általában elmondhatjuk, hogy egy 4 napos túra megtervezése is átlagosan 6 másodpercen belül tartható. Mivel már az első lépésben csökkentjük a gráf méretét, kijelölve a releváns csúcsok részhalmazát, ezért a napok számától (P) és a napi időkeretektől (T_{max}) függ a probléma mérete. Amennyiben egy olyan furcsa városban

végeznék a kísérletet, ahol minden csúcs meglátogatása pénzköltséggel jár, ez esetben a pénzügyi korlátunk is ugyanúgy lehet effektív, mint esetünkben az időkorlát. Az túrák megtervezésének átlagos számítási idejét a 8. ábrán foglaljuk össze. A tesztek az alábbi paraméterekkel rendelkező laptopon futtattuk: 3,8 GB RAM, Intel Core i3-3217U CPU, 1.80GHz \times 4 processzor. Minden paraméter kombinációra 20 alkalommal végeztünk el a tesztet, és az eredményeket (outlier értékektől való tisztítás nélkül) átlagoltuk. A futási idők érdekessége a 2- és 3 napos útvonalak tervezése közötti nagy eltérés az átlagos futási időben, míg ez nem növekszik 4 nap megtervezése esetén. Az a értékének változása látszólag nincs hatással a futási időre és a β paraméter is csak 3 és 4 napos túrák esetén növeli kis mértékben azt. Az α növekedése azonban 3 és 4 napos túrák esetén drasztikusan növeli a futási időt, mintegy 3-szorosára. Itt vélhetően az áll a háttérben, hogy ilyenkor egyre nehezebb olyan pontokat találnia az algoritmusnak, mely növelni tudná a célfüggvény értékét.

8. ábra: Az útvonaltervező algoritmus eredményei



- A napok kitöltöttsége csökkenő tendenciát mutat a napok számának növelésével (8. ábra). Az összes általunk vizsgált esetre átlagosan 94,3%-os értéket mértünk. A paraméterek közül csak az α paraméter növekedése van negatív hatással a napok kitöltöttségére, hiszen itt a célfüggvény hasznosság tényezője egyre kevésbé számít, így nehezebb olyan csúcsokat találni, melynek hasznossága ellensúlyozni tudja a látogatási idő - menetidő hányadosban bekövetkező romlást.
- A látogatási idő - menetidő hányados (mely a P napra együtt értendő) a napok számának növekedésével csökkenő tendenciát mutat, hiszen egyre nagyobb távolságra találjuk a következő csúcsokat, amelyeket még felvehetünk az utakba. Az a és β paraméter változása csekély hatással van a hányadosra, az α paraméter növelésével azonban drasztikusan növelhető a hányados értéke.

Empirikus vizsgálat

A kutatás jelen szakaszának egyik legfontosabb eredménye - mely egyben meg is különbözteti az összes eddig javasolt eljárástól - az a célfüggvény, mely reményeink szerint alkalmas arra, hogy jól, vagy az eddigiéknél jobban írja le a felhasználó céljait. Ezeket szem előtt tartva összehasonlítást végeztünk az általunk 4.4.4-es alfejezetben leírt eljárás és az utazóügynök probléma óta a területen széleskörben alkalmazott pontösszeg maximalizáló eljárás között. Tehát a két útvonaltervező algoritmus megegyezik a fent ismertetett algoritmussal, csupán a célfüggvényt változtatjuk. Az első esetben a saját paraméterbeállításaink szerint $\alpha = 0,75$ és $a = -1$ értékeket választjuk, míg a második esetben $\alpha = 0$ és $a = 0$ értékválasztással az alábbi célfüggvényhez jutunk:

$$C(\mathbf{0}, \beta, \mathbf{0}, R) = \sum_{p=1}^P \sum_{i=1}^N \theta_{ip} s_i$$

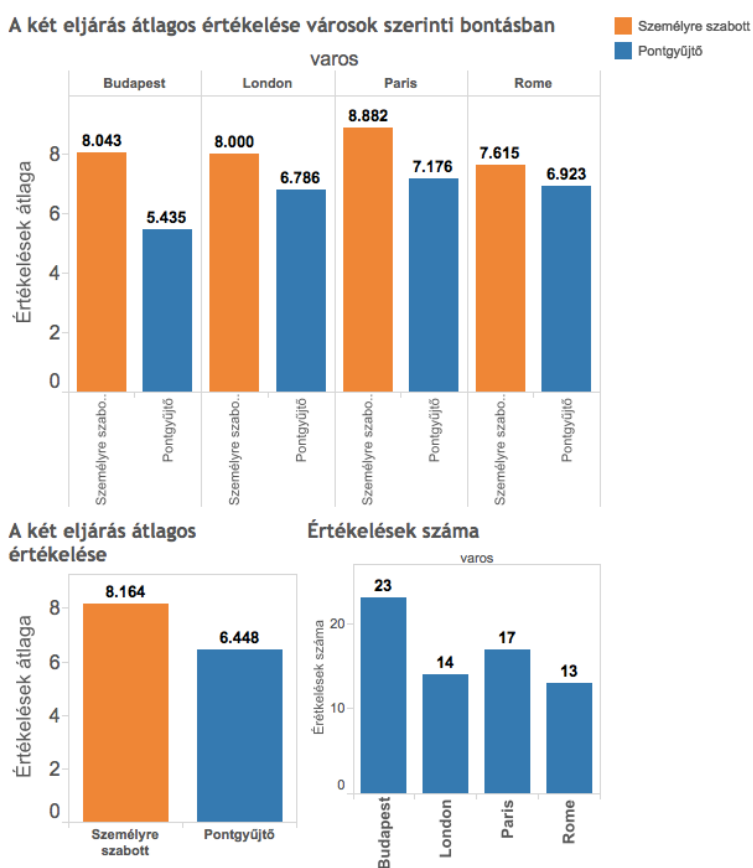
ami nem más, mint az utazóügynök problémából jól ismert célfüggvény, ahol minden meglátogatott csúcsban felvesszük az ott gyűjthető profitot, és célunk a túra végén a legnagyobb profitösszeg elérése. A vizsgálatához létrehoztunk egy a 3.8-as alfejezetben ismertetett vizsgálatához hasonló adatbázist és weboldalt. A felhasználók négy városra (Budapest, London, Párizs és Róma) tölthették ki a kérdőívet a travelscheduletest.hopto.org oldalon. A vizsgálat alapját képező nagyjából 950 látványosságot tartalmazó adatbázis struktúráját tekintve megegyezik az ajánlórendszer tesztelése során alkalmazottal, kiegészítve

a helyszíneken töltendő idővel (vizit idő), a helyszíneken fizetendő belépődíjakkal (amennyiben nem ingyenes), valamint a szélességi és hosszúsági értékekkel.

A felhasználók feladata, hogy a megadott preferenciáik és egyéb paraméterek alapján kalkulált 2 útvonaltervet (a tervbe felvett látványosságok és az őket összekötő útvonalak alapján) értékeljék 1-10-ig, ahol az 1-es a legkevésbé felel meg az ízlésüknek, míg a 10-es a számukra leginkább tetsző megoldást jelenti. A vizsgálat alá vont két eljárás közül azt értékeljük jobbnak, melyre a felhasználók a másikonál szignifikánsan magasabb értékelést adtak.

A kérdőívet 67 felhasználó töltötte ki, és az általuk adott értékelések alapján kijelenthetjük, hogy az általunk tervezett célfüggvénnyel kalkulált útvonalak átlagosan szignifikánsan jobb eredményt értek el. A 9. ábrán összefoglaljuk az értékelések végeredményét város szerinti bontásban és a teljes populációra nézve is.

9. ábra: A két eljárás által tervezett útvonalakra adott értékelések



Vizsgálatunk eredménye alapján kijelenthetjük, hogy az általunk tervezett célfüggvényt használó algoritmus szignifikánsan jobb útvonalakat generál a felhasználók számára, mint a korábban széles körben alkalmazott pontösszeg maximalizáló eljárás a teljes populációra nézve és minden városban külön-külön is. Ezzel elértük kezdeti célkitűzésünket, hiszen olyan személyre szabott útvonalakat tervező algoritmus megalkotása volt feladatunk, mely a felhasználóktól kapott minimális információ alapján képes a preferenciáinknak leginkább megfelelő ajánlásokat tenni. A fejezet zárásaként az alábbiakban a kutatás jelen szakaszának konklúzióját vonjuk le, valamint kijelöljük az előttünk álló fontosabb kutatási irányokat.

3. A kutatási eredmények összegzése

A dolgozatban az alábbi új tudományos eredmények kerültek bemutatásra:

1. Földfelszín modellek kidolgozása

Hogyan lehet a GPS alapú túranaplók adatait pontosabbá és elemzésre alkalmassá tenni? Létezik olyan digitális emelkedési modell (DEM), mely a valós magasság értékeket jól közelíti?

A GPS alapú túranaplók szélességi és hosszúsági adatainak korrekcióját Kálmán-filter alkalmazásával végeztük. Mivel a GPS magasság adatai nem kellő pontosságúak, így azokat egy digitális magasság modellből kell származtatnunk. A földfelszín közelítésére a NASA által közzétett adatok alapján **két DEM modellt is alkottunk: az egyik bilineáris interpoláción alapszik, és négyzetekkel közelíti a földfelszínt, míg a másik háromszögekkel teszi azt. A teszteredményeink alapján mindkét modell igen közeli eredményt ad a jelenlegi piaci sztenderdnek tekintett Google DEM modelljéhez.**

2. A túraszakaszok meredeksége és a túrázó sebessége közötti összefüggés meghatározása

Létezik a sebesség és az adott szakasz meredeksége közötti összefüggést leíró ismert megoldásoknál pontosabb?

A Waldo Tobler által 1993-ban közzétett tisztán meredekség alapú sebességbecslő eljárás óta nem született olyan becslési megoldás, mely ezt az összefüggést felülvizsgálná, pontosítaná. A rendelkezésünkre álló **2400 túranapló alapján illesztett meredekség-sebesség összefüggést leíró görbe eredményeink alapján jobb menetidőbecslést tesz lehetővé, mint a Tobler-görbe.** Kiemelendő, hogy az általunk tanulmányozott kutatások során használt adathalmazok mennyisége és minősége sem közelíti meg a vizsgálataink során felhasznált adathalmazét.

3. Menetidőbecslő eljárások megalkotása

Milyen eljárással adható a túrázók menetidejének egy a jelenlegieknél pontosabb becslése?

Menetidőbecslő eljárások évezredek óta léteznek, ám az elmúlt mintegy 120 évben a Naismith-féle ökölszabályt alkalmazták a gyakorlatban a túrázók menetidejének közelítésére. Ennek megannyi pontosítása született az idők folyamán, ennek legutóbbi mérföldköve Pittman et al. 2012-es eredménye, mely átlagosan 18%-os pontossággal becsül menetidőt túraútvonalakra. Az általunk javasolt két eljárás dinamikusan gyűjt adatot a túra közben és adaptálja azt a becslés során. **Az átlagsebességen alapuló eljárás átlagosan 12,94%-os pontosságú, míg a meredekség alapú eljárás 11,58%-os átlagos hibával működik** a vizsgált 2400 túranapló alapján. Ez az általunk ismert eddigi legpontosabb eljárás.

4. Turisztikai célú hibrid ajánlórendszer

Hogyan lehet kevés kezdeti információ alapján turisztikai célú ajánlórendszert építeni, mely személyre szabott ajánlásokat tesz a felhasználónak?

Mivel nem áll rendelkezésünkre a kutatás jelen fázisában egy olyan kiterjedt adatbázis, mely tartalmazná a felhasználók látványosságokra vonatkozó értékeléseit, így 3 európai fővárosra építettünk saját adatbázist, valamint egy 17 faktorból álló listát, melyek kombinációjával szeretnénk leírni az egyes helyszíneket (**tartalom alapú modul**). A **tudás alapú modul** a látványosságokat leíró faktorokra vonatkozó értékeléseket gyűjti be a felhasználótól, így a **tulajdonság kiterjesztő technikával**

integrált modulokból előálló hibrid rendszer képes ezen igen csekély kezdeti információból is személyre szabott ajánlásokat tenni.

5. A turisták klasszifikálása

Lehetséges a turistákat klasszifikálni a kezdetben magukkal kapcsolatban megadott információk alapján? Hogyan építhető fel ennek érdekében egy személyre szabott ajánlásokat adó rendszer kérdőíve?

Empirikus vizsgálatunkhoz egy weboldalt hoztunk létre, ahol az ajánlásokhoz csak az **általunk megalkotott 17 faktorra** kell értékelést adjanak a felhasználók. Ezek alapján **6 különböző turista típust sikerült azonosítanunk**, melyek közül az 3-3 hasonlóan rendezte sorba a felkínált 4 ajánlási eljárás eredményét. A csoportokról elmondható, hogy az őket leíró faktorok (melyekre jellemzően magas értékelést adtak) szoros összefüggésben vannak egymással, erre utal magas korrelációs együttható értékük. A turisták klasszifikálása segítséget nyújt az egyes felhasználók számára tett ajánlások további pontosításában.

6. Célfüggvény a TOP feladathoz

Milyen hasznosságfüggvénnyel írható le általános módon a felhasználók látványosságokra vonatkozó értékelése? Milyen célfüggvény vezet a gyakorlatban a felhasználók számára leginkább elfogadható útvonaltervezéshez?

Az útvonaltervező algoritmusok a TSP óta hagyományosan a csúcsokban gyűjthető profitok összegét tekintik az optimalizálandó célfüggvénynek, ez azonban figyelmen kívül hagy jópár gyakorlati megfontolást, éppen ezért ritkán vezet jó eredményre. Ilyen gyakorlati megfontolás például, hogy a felhasználó által alacsony értékelést kapott pontokat akkor se vegyük be az útvonaltervbe, ha azok igen kis költséggel megtehetőek, vagy éppen az, hogy igyekezzünk fajlagosan a lehető legtöbb időt a helyszínek meglátogatásával tölteni (a gráf élein történő séták helyett). **Javasolt hasznossági függvényünk és célfüggvényünk a korábbi pontösszeg-maximalizálás egy kiterjesztéseként értelmezhető**, hiszen a paraméterek bizonyos értékei mellett ($\alpha=0$) visszakapjuk azt:

$$u(s_i, a) = \begin{cases} \frac{1 - e^{-a(s_i - s^*)}}{a} & | a \neq 0 \\ s_i - s^* & | a = 0 \end{cases}$$

$$C(\alpha, \beta, a, R) = \left(\frac{\sum_{p=1}^P \sum_{i=1}^N \theta_{ip} v_i}{\sum_{i=1}^{N-1} \sum_{j=2}^N \tau_{ijp} t_{ij}^\beta} \right)^\alpha \times \left(\sum_{p=1}^P \sum_{i=1}^N \theta_{ip} u(s_i, a) \right)^{1-\alpha}$$

7. Heurisztikus algoritmus a TOP megoldására

Milyen algoritmus adható a kitűzött TOP feladatra, mely alacsony futási idejével alkalmazhatóvá teszi azt egy valós applikációban?

A TOP feladatra adott heurisztikus algoritmus célja az volt, hogy egyszerűségével, és ebből adódóan rövid futási idejével lehetőséget adjon annak későbbi gyakorlati alkalmazhatóságára. A **4 másodperc alatti eredmény**, valamint a célfüggvénynek köszönhető attraktív útvonaltervek megfelelő alapját képezik egy személyre szabott túrautakat tervező alkalmazás megalkotásának. Mivel a felhasználói elégedettség optimalizálását tartjuk legfőbb célunknak, így nagy sikerként könyvelhetjük el, hogy a felhasználók körében végzett tesztek alapján az általunk javasolt célfüggvény segítségével **szignifikánsan jobb útvonalterveket tudunk előállítani, mint a mások által használatos pontösszeg maximalizáló eljárással.**

4. Hivatkozások

- [1] D. Feillet - P. Dejax - M. Gandreau (2004): *Traveling Salesman Problems with Profits*, Journal of Transportation Science, Vol. 39, No. 2, pp. 188-205. DOI: 10.1287/trsc.1030.0079
- [2] K. Sylejmani - A. Dika (2011): *Solving touristic trip planning problem by using taboo search approach*, International Journal of Computer Science Issues, Vol. 8, Issue 5, No 3, pp. 139-149. doi: 10.1109/HIS.2012.6421351
- [3] W. Naismith: *Notes and queries*, [1892] Scottish Mountaineering Club Journal, Vol. 2, p. 133.
- [4] E. Langmuir (1995): *Mountaineering and Leadership*, 3rd ed. Sportscotland.
- [5] R. Aitken (1977): *Wilderness Areas in Scotland*, unpublished Ph.D. Thesis. University of Aberdeen. Aberdeen.
- [6] W. Tobler (1993): *Three presentations on geographical analysis and modeling: Non-isotropic geographic modeling speculations on the geometry of geography global spatial analysis*, National Center for Geographic Information and Analysis Technical Report, Vol. 93, No. 1, pp. 1–24.
- [7] A. Pitman - M. Zanker - J. Gamper - P. Andritsos (2012): *Individualized hiking time estimation*, in Proceedings of the 23rd International Workshop on Database and Expert Systems Applications, pp. 101-105. doi: 10.1109/DEXA.2012.51
- [8] M.H. Ferrara - M. P. LaMeau (2012): *Pandora Radio/Music Genome Project. Innovation Masters: History's Best Examples of Business Transformation*. Detroit. pp. 267-270. Gale Virtual Reference Library. doi: 10.5860/CHOICE.50-2756
- [9] G. Linden - B. Smith - J. York (2003): *Amazon.com Recommendations: Item-to-Item Collaborative Filtering*, IEEE Internet Computing, Vol. 7, No. 1, pp. 76-80. doi: 10.1109/MIC.2003.1167344
- [10] R. Burke (2000): *Knowledge-based Recommender Systems*, Encyclopedia of Library and Information Science, Vol. 69, No. 32, pp. 180-200. doi:10.1.1.21.6029&rank=1

- [11] P. Melville - R.J. Mooney - R. Nagarajan (2002): *Content-Boosted Collaborative Filtering for Improved Recommendations*, in Processing of the 18th National Conference on Artificial Intelligence, pp. 187-192. doi: 10.1109/CSNT.2012.218
- [12] P. Vansteenwegen - W. Souffriau - G. Vanden Berghe - D.D. Van Oudheusden (2011): *The city trip planner: an expert system for tourists*, Expert Systems with Applications, Vol. 38. No. 6, pp. 6540–6546. doi:10.1016/j.eswa.2010.11.085
- [13] <http://e4ftl01.cr.usgs.gov/SRTM/SRTMGL1.003/2000.02.11/>
- [14] C. Miller - A. Tucker - R. Zemlin (1960): *Integer programming formulations and travelling salesman problems*, Journal of the ACM, Vol. 7, pp. 326–329. doi: 10.1145/321043.321046
- [15] J.A. Hartigan - M.A. Wong (1979): *Algorithm AS 136: A K-Means Clustering Algorithm*, Journal of the Royal Statistical Society, Series C, Vol. 28, No. 1, pp. 100-108. DOI: 10.2307/2346830
- [16] G. Gutin - A. Yeo - A. Zverovich (2002): *Traveling salesman should not be greedy: domination analysis of greedy-type heuristics for the TSP*, Discrete Applied Mathematics, Vol. 117, pp. 81-86. doi:10.1016/S0166-218X(01)00195-0

5. Saját publikációk a témakörben

Referált szakmai folyóiratok magyar nyelven

Apáthy M. Sándor [2016]: *Egy heurisztikus útvonaltervező algoritmus többnapos túrák tervezésére*, Szigma Matematikai-közgazdasági folyóirat, Vol. 3-4, elfogadva

Apáthy M. Sándor [2016]: Az útvonaltervező algoritmusok történeti áttekintése, különös tekintettel azok turisztikai célú alkalmazásaira, Alkalmazott Matematikai Lapok, Vol. 33, elfogadva

Apáthy M. Sándor [2016]: A menetidőbecslés alkalmazásai, Közlekedéstudományi Szemle, elfogadva

Apáthy M. Sándor [2016]: Turistatípusok azonosítása - Egy lehetséges turisztikai ajánlórendszer, Vezetéstudomány, bírálat alatt

Egyéb publikációk magyar nyelven

Apáthy M. Sándor [2016]: *Földfelszíni gyalogos közlekedés modellezése*, Innováció és fenntartható felszíni közlekedés Konferencia 2016. Budapest, Magyarország, 2016.08.29 - 2016.08.31, paper 22, ISBN: 978-963-88875-2-8

Apáthy M. Sándor [2011]: Fejezetek a modern közgazdaságtudományból – recenzió [Móczár József: Fejezetek a modern közgazdaságtudományból. Akadémiai Kiadó, Budapest. 2008. 608 oldal ISBN 9789630585378], Gazdaság és társadalom, Vol. 3, No. 3-4, pp. 189-194. ISSN 0865-7823

Referált szakmai folyóiratok idegen nyelven

Apáthy M. Sándor [2016]: *Personalised hiking time estimation*, Pure Mathematics and Applications, megjelenés alatt

Apáthy M. Sándor [2016]: *Practical Route Planning Algorithm*, Periodica Polytechnica Transportation Engineering, Vol 45, No. 2, elfogadva