Expanding Reverse Nearest Neighbors

Wentao Li
The Hong Kong University of Science and Technology (Guangzhou)
wentaoli@hkust-gz.edu.cn

Maolin Cai
Chongqing University
caimaolin@cqu.edu.cn

Min Gao
Chongqing University
gaomin@cqu.edu.cn

Dong Wen
The University of New South Wales
dong.wen@unsw.edu.au

Lu Qin
AAII, FEIT, University of Technology Sydney
lu.qin@uts.edu.au

Wei Wang*
The Hong Kong University of Science and Technology (Guangzhou)
The Hong Kong University of Science and Technology
weiwcs@ust.hk

Absztrakt

Egy gráfban egy csúcs fordított legközelebbi szomszédjai (reverse nearest neighbors, RNN) azon csúcsok halmazát jelentik, amelyek ezt a csúcsot tekintik a legközelebbi szomszédjuknak. Amikor például egy csúcs egy létesítményt képvisel, mint egy metróállomást, akkor annak RNN-jébe azok a potenciális felhasználók tartoznak, akik ezt a legközelebbi létesítményt részesítik előnyben. Gyakran előfordulhatnak alulhasznált létesítmények kis RNN-mérettel, és ezeknek a létesítményeknek az áthelyezése a nagyobb kiszolgálás érdekében költséges vagy kivitelezhetetlen lehet. Egy költséghatékonyabb megoldás bizonyos élek szelektív javítását foglalja magában (pl. a súlyaik csökkentésével), hogy így növeljük az alulhasznált létesítmények RNN-méretét. Ez vezette a kutatásunkat az RNN bővítésének problémájához (Expanding Reverse Nearest Neighbors, ERNN), amelynek célja a cél létesítmény RNN-méretének maximalizálása korlátozott számú él fejlesztésével. Az ERNN-probléma megoldása lehetővé teszi, hogy az alulhasznált létesítmények több felhasználót szolgáljanak ki, enyhítve a terheket más létesítményeken. Számos lehetséges alkalmazás ellenére az ERNN-probléma nehezen megoldható: bizonyíthatóan NP-nehéz és APX-nehéz, továbbá nem monoton és nem szubmoduláris tulajdonságokat mutat. E kihívások leküzdésére új, mohó algoritmusokat javaslunk, amelyek a szükséges élek számának és az egyes élek feldolgozási költségének minimalizálásával javítják a hatékonyságot. Kísérleti eredményeink azt

mutatják, hogy a javasolt algoritmusok nagyságrendekkel gyorsabbak a szokványos mohó algoritmushoz képest, miközben jelentősen növelik az RNN méretét.

Bevezetés

A gráfok széles körben használatosak különféle entitások és azok kapcsolatai ábrázolására. Jelen cikk az úthálózatokra koncentrál, amelyek egy típusos súlyozott gráfként foghatók fel. Az úthálózatban az egyes helyeken elhelyezkedő entitásokat csúcsok, míg az entitások közötti utakat élek képviselik, ahol az utak mentén mért utazási idő az élsúlyokat jelenti. Bizonyos alkalmazásokban, például döntéstámogató rendszerekben, az úthálózat csúcsai két kategóriába sorolhatók: felhasználók (C) és létesítmények (F). A legtöbb esetben a felhasználók ($c \in C$) a legközelebbi létesítményt választják az igényelt szolgáltatások elérése érdekében. Ha egy létesítmény csúcs ($f \in F$) összes felhasználóját összegyűjtjük, akik f-et tekintik legközelebbi létesítményüknek, akkor megkapjuk f létesítmény (bikromatikus) fordított legközelebbi szomszédjait (RNN). Ilyen módon a f létesítmény RNN-je azon potenciális felhasználókat tartalmazza, akik számára f a legpreferáltabb választás a szolgáltatások elérésére. Az RNN fogalma számos alkalmazási területen megjelenik: például az üzleti életben, éttermek RNN-elemzése segíthet azonosítani a célzott marketingkampányok potenciális ügyfeleit, míg a közlekedéstervezésben a közlekedési csomópontok RNN-jének elemzése támogatja az új állomások helyének meghatározását.

Motiváció

Egy létesítmény RNN-jének mérete bizonyos fokig annak kihasználtságát mutatja, mivel az RNN azokat a felhasználókat foglalja magában, akik jellemzően az adott létesítményt választják. Az RNN eredeti célja éppen a létesítmények hatókörének és kihasználtságának mérése volt. Azonban azt tapasztaltuk, hogy nem minden létesítmény teljesen kihasznált; egyes létesítmények esetében kis RNN-méret figyelhető meg, jellemzően távoli elhelyezkedésük miatt. Ez erőforráspazarláshoz vezethet, mivel az alulhasznált létesítmények nem használják ki teljes szolgáltatási potenciáljukat, és így további terhet jelentenek más létesítmények számára. Az alulhasznált létesítmények kihasználtságának növelésére az egyik megoldás az áthelyezésük lenne, ám ez költséges, vagy akár megvalósíthatatlan lehet, különösen tűzoltóállomások esetében. Egy költséghatékonyabb alternatíva a létesítmények kihasználtságának javítására egy korlátozott számú útszakasz (azaz egy költségvetés) kiválasztása és korszerűsítése az élsúlyok csökkentésével. Ez megvalósítható gyorsforgalmi sávok vagy szélesebb utak építésével, amelyek az élek súlyát csökkentik. Míg az útkorszerűsítéssel kapcsolatos kutatások eddig a hálózati átmérő vagy a késleltetés csökkentésére irányultak, a létesítmények kihasználtságának javítását célzó vizsgálatok még nem születtek. Ez motiválta az ERNN-probléma, azaz az RNN bővítésének problémájának vizsgálatát úthálózatokban: egy gráf (G), egy cél létesítmény (f) és egy költségvetés (b) adott; célunk, hogy f létesítmény RNN-méretét maximalizáljuk legfeljebb b él súlyának csökkentésével.

Az ERNN-probléma alkalmazásai

- **Közösségi közlekedés tervezése:** A környezeti és gazdasági szempontok miatt a közösségi közlekedés, mint a metró, egyre elterjedtebb. Azonban néhány metróállomás kis RNN-mérettel rendelkezik, ami alacsony kihasználtságot eredményez. Az új állomások helyének megválasztása vagy új állomások építése költséges és akár kivitelezhetetlen is lehet. Egy járható út az ERNN-probléma megoldása, amellyel bizonyos utak fejlesztése révén növelhetjük ezen állomások RNN-jét, javítva az alulhasznált állomások kihasználtságát, enyhítve a terhelést más állomásokon.
- **Vészhelyzeti reagálás tervezése:** Az olyan vészhelyzeti létesítmények, mint kórházak vagy tűzoltóságok, alapvetőek az emberek biztonsága szempontjából. Azonban előfordulhat, hogy egyes létesítmények kis RNN-mérettel rendelkeznek, és nem tudják teljes mértékben kihasználni kapacitásaikat vészhelyzetek során, ami erőforráspazarlást eredményez. Az ilyen létesítmények kihasználtságának növelésére egy megoldás az ERNN-probléma megoldása bizonyos utak fejlesztésével, így növelve a létesítmény

RNN-méretét. Ez nem csupán elkerüli az alulhasznált létesítmények áthelyezésének vagy új létesítmények építésének költségeit, hanem gyorsabb vészhelyzeti reagálást tesz lehetővé az útfejlesztések révén csökkentett utazási idő által.

Kapcsolódó munkák

RNN-számítás

Korábbi kutatások főként a gráfok csúcspontjainak RNN-számítására koncentráltak metrikus terekben. Yiu és társai bevezették az RNN-számítást gráfokban, beleértve a bikromatikus RNN problémát (ahogyan ebben a cikkben is alkalmazzuk, ahol a csúcsok felhasználókra és létesítményekre oszlanak) és a monokromatikus RNN problémát (ahol a gráf egyetlen típusú csúcsból áll). Safer és társai a Voronoi-diagramot használták az RNN-számításhoz, míg Efentakis és társai hub címkéket alkalmaztak a számítás gyorsítására.

A kutatások foglalkoznak az RNN-ek karbantartásával is. Például Sun és társai, valamint Li és társai olyan úthálózat-indexelési módszereket javasoltak, amelyek lehetővé teszik az RNN-frissítéseket egy lekérdezési csúcs esetén. Cheema és társai egy szűrés-hitelesítési keretrendszert vezettek be az RNN-frissítések kezelésére. Azonban, tudomásunk szerint, ezek a kutatások főként arra fókuszálnak, hogyan lehet az RNN-t figyelemmel kísérni, amikor a létesítmények/felhasználók helyzetet változtatnak. További kutatás szükséges arra vonatkozóan, hogy miként lehet frissíteni egy lekérdezési csúcs RNN-jét, ha a hálózat súlyai változnak.

Egy másik kapcsolódó téma a fordított legközelebbi szomszédok maximalizálási problémája, amelynek célja egy új létesítmény optimális helyének meghatározása, hogy maximalizálja annak RNN-méretét. Ezzel szemben az általunk javasolt ERNN probléma aktívan a gráf éleinek fejlesztésére irányul, hogy maximalizálja egy adott létesítmény RNN-méretét. Ennek megfelelően az általunk vizsgált probléma eltér a létező problémáktól, és további kutatást igényel.

Hálózatfejlesztés

A jelen tanulmányban vizsgált ERNN probléma a hálózatfejlesztési problémák közé tartozik, ahol a cél hálózati élek vagy csúcsok fejlesztésével egy adott célkitűzés optimalizálása. A hálózatfejlesztési problémák kutatása hosszú múltra tekint vissza. Például Paik és társai 1995-ben kezdtek el definiálni különféle hálózatfejlesztési problémákat, amelyek célja specifikus hálózati mutatók, például a legrövidebb vagy leghosszabb távolság optimalizálása volt. Zhang és társai az él-súlyok csökkentésére összpontosítottak a csúcsok közötti távolságok mérséklése érdekében. Hasonlóképpen, Campbell és társai a leghosszabb utazási idő minimalizálását tanulmányozták egyes élek súlyának csökkentésével. Medya és társai módszereket vizsgáltak a hálózati késleltetés minimalizálására a csúcs-súlyok nullára csökkentésével. Lin és társai szintén az egyes csúcshalmazok közötti összes távolság minimalizálását tanulmányozták élek súlyának csökkentésével. Ezzel szemben az ERNN probléma célja egy céllétesítmény RNN-méretének maximalizálása, amely új megoldások fejlesztését igényli.

Egyéb kapcsolódó problémák közé tartozik élek hozzáadása egy gráfhoz (ami felfogható úgy is, mint a súlyok végtelenről nullára csökkentése), hogy minimalizálják a gráf átmérőjét, maximalizálják az információterjedést, növeljék a k-magot vagy a k-trusst, fokozzák a centralitást, vagy optimalizálják a klaszterezést. Tekintettel arra, hogy egy úthálózatban az új utak létrehozása nem mindig kivitelezhető, ebben a kutatásban nem foglalkozunk új élek hozzáadásával. Azonban ez érdekes irány lehet a jövőbeli kutatások számára.

Kihívások

Az ERNN-probléma megoldása komoly számítási nehézségekkel jár. Egy gráfban lévő létesítmény RNN-jének növelése érdekében számos útvonal kombinációját kell megvizsgálni, amelyek potenciálisan növelhetik a létesítmény vonzáskörzetét. A probléma NP-nehézsége azt jelenti, hogy egy egzakt megoldás megtalálása nagy hálózatok esetében nem lehetséges ésszerű időn belül.

Továbbá a probléma nem monotonnak és nem szubmodulárisnak bizonyul, ami azt jelenti, hogy a standard közelítő algoritmusok nem működnek megfelelően. Ennek ellenére a mohó algoritmusok képesek közel optimális megoldásokat találni a gyakorlatban, feltéve, hogy megfelelő technikákat alkalmaznak a számítási idő csökkentésére.

Alapfogalmak

A cikk alapfogalmainak megértéséhez íme a legfontosabb elemek összefoglalása:

1. **Gráfok és úthálózatok**

- **Gráfok**: A gráfok különféle entitások és azok közötti kapcsolatok ábrázolására szolgálnak. Egy **úthálózat** – ami egy súlyozott gráf típusa – esetében a csúcsok konkrét helyszíneket (pl. metróállomások vagy felhasználói pontok) jelképeznek, míg az élek az ezen helyszínek közötti utakat reprezentálják. Az éleken lévő súlyok a két hely közötti utazási időt jelölik, ami támogatja az ilyen hálózatok alkalmazását közlekedési és létesítménytervezési célokra.

2. **Felhasználói és létesítményi csúcsok**

- **Felhasználók (C)** és **létesítmények (F)**: Az úthálózat csúcsai két kategóriába sorolhatók: felhasználók (C), akik szolgáltatásokat keresnek, és létesítmények (F), amelyek ezeket a szolgáltatásokat nyújtják. A felhasználók rendszerint a legközelebbi létesítményt választják az igényelt szolgáltatások eléréséhez, így a felhasználók és létesítmények közötti kapcsolat alapvetően távolságfüggő.

3. **Fordított legközelebbi szomszédok (RNN)**

- **Definíció**: Egy létesítményi csúcs (f) fordított legközelebbi szomszédjai (RNN) azoknak a felhasználói csúcsoknak a halmaza,

amelyek számára f a legközelebbi létesítmény. Az RNN fogalma különösen fontos egy létesítmény hatókörének vagy befolyásának meghatározásában, mivel az RNN mérete a létesítmény iránti keresletet tükrözi.
- **Alkalmazások**: Az RNN-t számos területen alkalmazzák, például a célzott marketingben, ahol a legközelebbi létesítmény alapján segíthet potenciális ügyfelek azonosításában, illetve közlekedéstervezésben, ahol egy állomás RNN-jének elemzése feltárhatja a szolgáltatásai iránti keresletet.

4. Az Expanding Reverse Nearest Neighbors (ERNN) problémája

- **Indíttatás**: Egyes létesítmények kis RNN-mérettel rendelkezhetnek, például távoli elhelyezkedésük miatt, ami alacsony kihasználtsághoz és erőforráspazarláshoz vezethet. Az alulhasznált létesítmények áthelyezése költséges és gyakran kivitelezhetetlen, így felmerül az igény költséghatékony megoldásokra, melyek az RNN méretét növelik.
- **Cél**: Az ERNN probléma célja egy céllétesítmény RNN-méretének maximalizálása meghatározott számú él fejlesztésével, amely során az élek súlyait csökkentik. Ezzel a módszerrel az utazási idő csökken, így a létesítmény hatóköre és felhasználói bázisa növekszik.
- **Kihívások**: Az ERNN probléma NP-nehéz és APX-nehéz, ami azt jelenti, hogy pontos megoldása nagyon összetett. Továbbá, a probléma nem mutat monoton és szubmoduláris tulajdonságokat, így a hagyományos mohó algoritmusok nem alkalmazhatók egyszerűen.

5. Algoritmusok az ERNN probléma megoldására

- **Mohó algoritmus**: Az ERNN probléma megoldásához egy mohó algoritmust alkalmaznak, amely iteratív módon választja ki azokat az élfejlesztéseket, amelyek a legnagyobb mértékben növelik az RNN méretét. Azonban a standard mohó algoritmus minden lehetséges élkombinációt kiértékel, ami időigényes.
- **Optimalizálási technikák**:

- **Távolságalapú élvizsgálat**: Ez a technika a csúcsokat a céllétesítményhez való távolság alapján rendezi, és csak azokat az éleket vizsgálja, amelyek valószínűleg nagyobb hatással vannak az RNN-re, ezzel elkerülve a távolabbi, valószínűleg kevésbé hatékony élek feldolgozását.
- **Inkrementális RNN-számítás**: Ez a megközelítés elkerüli az egész RNN újraszámítását minden egyes élmódosítás után, és csak az RNN méretében bekövetkező változásokat számolja ki, ami jelentős időmegtakarítást eredményez.

Eredmények

Az ERNN probléma formálisan a következőképpen definiálható: Adott egy gráf G(V,E,W), ahol V a csomópontok (felhasználók és létesítmények), E az élek (utak), W az élek súlyai (utazási idők) jelöli. A feladat az, hogy a költségvetésnek megfelelő számú élt válasszuk ki és fejlesszük úgy, hogy a cél létesítmény RNN-mérete maximális legyen.

Az egyszerűsített változatban az élek súlyait nullára lehet csökkenteni, vagyis az utak fejlesztése gyakorlatilag ingyenes hozzáférést biztosít a cél létesítményhez. Ez a leegyszerűsítés lehetővé teszi az algoritmusok hatékonyabb alkalmazását, anélkül, hogy az általános megoldhatóság veszélybe kerülne.

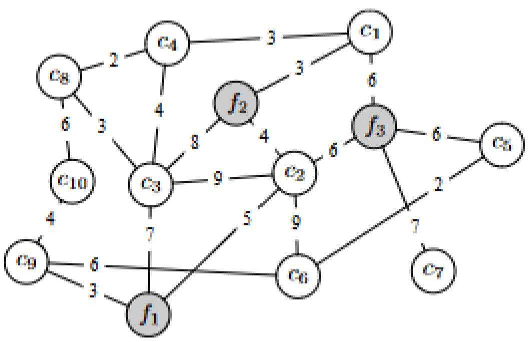


Figure 1: The Example Graph G

Algorithm 1: Yiu's Algorithm

Input: graph $G(V = F \cup C, E, W)$, facility $f \in F$

Output: $\mathrm{RNN}_G(f)$

$processed(v) \leftarrow 0, \mathrm{NN}(v) \leftarrow -1, dist(v) \leftarrow \infty$, for $\forall v \in V$;
$Q \leftarrow \emptyset$;
$\mathrm{NN}(p) = p, dist(p) \leftarrow 0$, for $\forall p \in F$;
push $\{p, dist(p), \mathrm{NN}(p)\}$ into Q, for $\forall p \in F$;
while *Q is not empty* **do**
    pop $\{u, dist(u), \mathrm{NN}(u)\}$ from Q;
    if $processed(u) = 1$ **then** continue;
    $processed(u) \leftarrow 1$;
    for *each $v \in N(u)$ and $processed(v) \neq 1$* **do**
        if $dist(u) + w(u, v) < dist(v)$ **then**
            $\mathrm{NN}(v) \leftarrow \mathrm{NN}(u), dist(v) \leftarrow dist(u) + w(u, v)$;
            push $\{v, dist(v), \mathrm{NN}(v)\}$ into Q;
for *each $v \in C$* **do**
    if $\mathrm{NN}(c) = f$ **then** add c into $\mathrm{RNN}(f)$;
return $\mathrm{RNN}(f)$;

Mohó algoritmus és fejlesztések

A cikkegy új mohó algoritmust mutat be, amely két fő technikával javítja az ERNN-probléma megoldásának hatékonyságát:

1. **Távolságalapú élvizsgálat**: Az algoritmus az éleket a cél létesítménytől való távolságuk alapján rendezi. Az éleket ebben a sorrendben vizsgálja, és minden élhez kiszámít egy felső határt az elérhető RNN növekedésre. Ha a távolabbi élek fejlesztése már nem eredményez nagyobb növekedést, az algoritmus abbahagyja ezek vizsgálatát, és nem pazarolja az erőforrásokat olyan élekre, amelyek nem járulnak hozzá a cél eléréséhez.
2. **Inkrementális RNN-számítás**: Minden egyes él fejlesztése után csak az érintett részeket számolja újra az algoritmus, ahelyett, hogy minden iterációban újraszámítaná az egész gráfot. Ez jelentősen csökkenti a számítási költségeket és felgyorsítja a megoldást.

Az algoritmus ezen fejlesztései lehetővé teszik, hogy nagy gráfok esetén is hatékony megoldást találjunk, minimalizálva a számítási időt.

Algorithm 2: Basic

Input: graph $G(V = F \cup C, E, W)$, modifiable edges $M \subseteq E$, budget b, target facility $f \in F$

Output: edges $A \subseteq M$

while *budget* $b > 0$ **do**
    $opt \leftarrow 0$;
    `// Distance-Based Edge Inspection (Sec. 4.2)`
    for *each edge* $e \in M$ **do**
        $G' \leftarrow$ upgrade e in G;
        `// Incremental RNN Computation (Sec. 4.3)`
        compute $\mathrm{RNN}_{G'}(f)$ in G' using Algorithm 1;
        if $opt > |\mathrm{RNN}_{G'}(f)|$ **then**
            $opt \leftarrow |\mathrm{RNN}_{G'}(f)|$, $ans \leftarrow e$;
    $M \leftarrow M \setminus ans$, $A \leftarrow A \cup ans$;
    $G \leftarrow$ upgrade ans in G, $b \leftarrow b - 1$;
return *edges* A;

Algorithm 3: DBEI

Input: graph $G(V = F \cup C, E, W)$, modifiable edges $M \subseteq E$, budget b, target facility $f \in F$

Output: edges $A \subseteq M$

while *budget* $b > 0$ **do**
    $opt \leftarrow 0$;
    $dist(v) \leftarrow -1, visited(v) \leftarrow 0$, for $\forall v \in V$;
    push f into Q;
    while Q *is not empty* **do**
        pop u from Q;
        $visited(u) \leftarrow 1$;
        for $v \in N(u)$ *and* $visitd(v) = 0$ **do**
            if $dist(u) + w(u, v) < dist(v)$ **then**
                $dist(v) \leftarrow dist(u) + w(u, v)$;
                push v into Q;
            if $e = (u, v) \in M$ *and not used* **then**
                $dist((e), f) \leftarrow dist(u)$;
                $\Delta_e \leftarrow \{c \notin \mathrm{RNN}_G(f) | dist_G(c, \mathrm{NN}_G(c)) > dist((e), f)\}$;
                $ub(e) \leftarrow |\Delta(e) \cup \mathrm{RNN}_G(f)|$;
                if $ub(e) \leq opt$ **then** stop this round;
                compute $\mathrm{RNN}_{G'}(f)$ in G' using Algorithm 1;
                if $opt > |\mathrm{RNN}_{G'}(f)|$ **then**
                    $opt \leftarrow |\mathrm{RNN}_{G'}(f)|$, $ans \leftarrow e$;
    $M \leftarrow M \setminus ans$, $A \leftarrow A \cup ans$;
    $G \leftarrow$ upgrade ans in G, $b \leftarrow b - 1$;
return *edges* A;

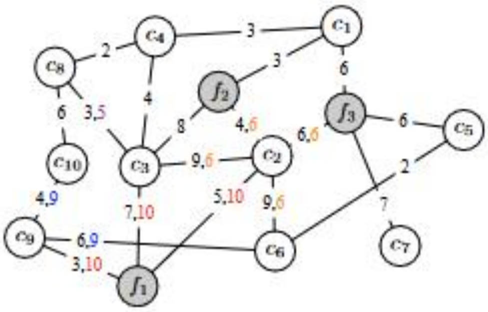
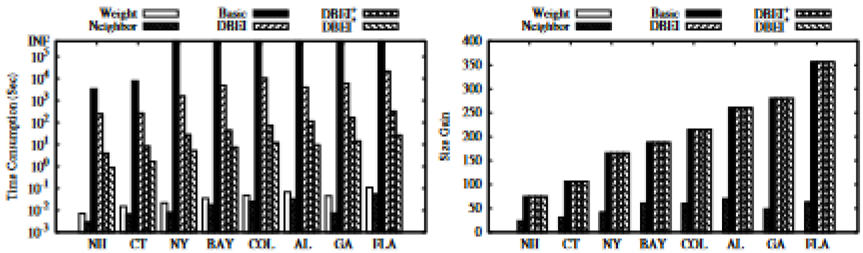


Figure 3: The Execution Process of Algorithm 3

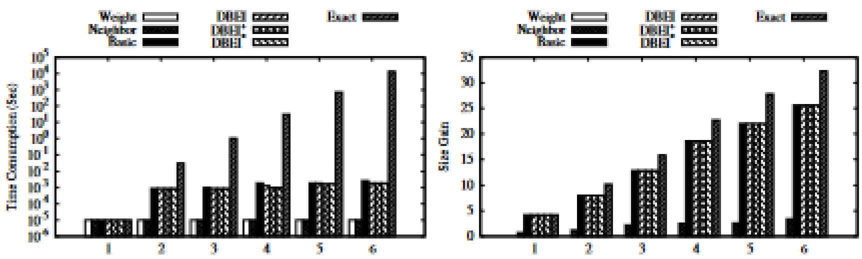
Table 1: Description of Datasets

Name	$\lvert V \rvert$	$\lvert E \rvert$
NH	116,920	133,415
CT	153,011	187,318
NY	264,346	366,923
BAY	321,270	400,086
COL	435,666	528,533
AL	566,843	661,487
GA	738,879	869,890
FLA	1,070,376	1,356,399



(a) The Comparison of the Running Time (b) The Comparison of the RNN Size Gain

Figure 4: The Comparison Among Various Methods



(a) The Comparison of the Running Time (b) The Comparison of the RNN Size Gain

Figure 5: The Comparison with the Exact Method

Kísérleti eredmények

A kutatók kísérleteket végeztek valós közúti hálózatok felhasználásával, hogy értékeljék a javasolt algoritmus teljesítményét. Az eredmények azt mutatják, hogy a javasolt mohó algoritmus jelentősen gyorsabb, mint a hagyományos módszerek. Például képes több millió élt tartalmazó hálózatokat megoldani kevesebb, mint egy perc alatt, ami kiemelkedő teljesítménynek számít.

Az algoritmust egy egzakt algoritmussal is összehasonlították, és kiderült, hogy az új algoritmus által elért RNN növekedés közel megegyezik az optimális megoldással. Ez azt jelenti, hogy a javasolt módszer mind hatékony, mind pedig eredményes, és alkalmazható valós hálózatokon is.

Alkalmazási területek

- **Közösségi közlekedés**: A kis RNN-méretű metróállomások RNN-jének bővítése növelheti kihasználtságukat, és egyenletesebben oszthatja el a terhelést az állomások között, ezáltal csökkentve a zsúfoltságot a forgalmasabb állomásokon.
- **Vészhelyzeti reagálás**: Az olyan vészhelyzeti létesítmények, mint a kórházak vagy a tűzoltóságok RNN-jének növelése javítja az elérhetőséget, és gyorsabb reagálást tesz lehetővé válsághelyzetekben, anélkül, hogy költséges áthelyezésekre vagy új létesítmények építésére lenne szükség.

Következtetések

A cikk egy új megközelítést kínál a közlekedési és más hálózati rendszerek optimalizálására, különösen a létesítmények kihasználtságának javítására. Az ERNN-probléma megoldása a létesítmények fordított legközelebbi szomszédjainak növelésére koncentrál, és az algoritmusok hatékonyan alkalmazhatók valós problémák megoldására is. A kísérleti eredmények igazolják, hogy az új algoritmusok gyorsak és hatékonyak, és jelentős előrelépést jelentenek a nagy hálózatok optimalizálásában.

Irodalomjegyzék

[1] Nasser Allheeib, Kiki Adhinugraha, David Taniar, and Md Saiful Islam. 2022. Computing reverse nearest neighbourhood on road maps. World Wide Web(2022), 1–32.

[2] Elisabetta Bergamini, Pierluigi Crescenzi, Gianlorenzo D'angelo, Henning Mey-erhenke, Lorenzo Severini, and Yllka Velaj. 2018. Improving the betweennesscentrality of a node by adding links. Journal of Experimental Algorithmics (JEA)23 (2018), 1–32.

[3] Ann Melissa Campbell, Timothy J Lowe, and Li Zhang. 2006. Upgrading arcsto minimize the maximum travel time in a network. Networks: An International Journal 47, 2 (2006), 72–80.

[4] Vineet Chaoji, Sayan Ranu, Rajeev Rastogi, and Rushi Bhatt. 2012. Recommen-dations to boost content spread in social networks. In Proceedings of the 21st international conference on World Wide Web. 529–538.

[5] Muhammad Aamir Cheema, Wenjie Zhang, Xuemin Lin, Ying Zhang, and Xuefei Li. 2012. Continuous reverse k nearest neighbors queries in euclidean space and in spatial networks. The VLDB Journal 21 (2012), 69–95.

[6] Farhana M Choudhury, J Shane Culpepper, Timos Sellis, and Xin Cao. 2016. Maximizing bichromatic reverse spatial and textual k nearest neighbor queries. Proceedings of the VLDB Endowment 9, 6 (2016), 456–467.

[7] Edsger W. Dijkstra. 1959. A note on two problems in connexion with graphs. Numer. Math. 1 (1959), 269–271. https://doi.org/10.1007/BF01386390

[8] Alexandros Efentakis and Dieter Pfoser. 2016. ReHub: Extending hub labels for reverse k-nearest neighbor queries on large-scale networks. Journal of Experimental Algorithmics (JEA) 21 (2016), 1–35.

[9] Wenfei Fan. 2022. Big graphs: challenges and opportunities. Proceedings of the VLDB Endowment 15, 12 (2022), 3782–3797.

[10] Richard M Karp. 1972. Reducibility among combinatorial problems. In Complexity of computer computations. Springer, 85–103.

[11] Flip Korn and Suresh Muthukrishnan. 2000. Influence sets based on reverse nearest neighbor queries. ACM Sigmod Record 29, 2 (2000), 201–212.

[12] Guohui Li, Yanhong Li, Jianjun Li, LihChyun Shu, and Fumin Yang. 2010. Contin- uous reverse k nearest neighbor monitoring on moving objects in road networks. Information Systems 35, 8 (2010), 860–883.

[13] Wentao Li, Maolin Cai, Min Gao, Dong Wen, Lu Qin, and Wei Wang. 2023. Technical Report. https://www.dropbox.com/scl/fo/7jkd2bgwzvg5rv8bj2fmr/h?rlkey=673fyrek4rlvq10q4oyuva4ql&dl=0

[14] Wentao Li, Min Gao, Dong Wen, Hongwei Zhou, Cai Ke, and Lu Qin. 2022. Manip- ulating Structural Graph Clustering. In 2022 IEEE 38th International Conference on Data Engineering (ICDE). IEEE, 2749–2761.

[15] Wentao Li, Min Gao, Fan Wu, Wenge Rong, Junhao Wen, and Lu Qin. 2021. Manipulating black-box networks for centrality promotion. In 2021 IEEE 37th International Conference on Data Engineering (ICDE). IEEE, 73–84.

[16] Xueping Li, Zhaoxia Zhao, Xiaoyan Zhu, and Tami Wyatt. 2011. Covering models and optimization techniques for emergency response facility location and planning: a review. Mathematical Methods of Operations Research 74 (2011), 281–310.