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

Bevezetés

A fordított legközelebbi szomszédok (RNN) problémája széleskörűen alkalmazható a gráf alapú rendszerekben, különösen közúti hálózatokban. A fordított legközelebbi szomszéd (RNN) egy létesítményhez tartozó azon felhasználók halmazát jelenti, akik azt a létesítményt választják legközelebbi elérhető szolgáltatásként. Ez különösen fontos az olyan rendszerekben, mint a közösségi közlekedés vagy a vészhelyzeti reagálás, ahol a létesítmények hozzáférhetősége és kihasználtsága kulcsfontosságú. Az RNN-k nagysága közvetlenül befolyásolja a létesítmények hatékonyságát, mivel a nagyobb RNN több felhasználót jelent, akik az adott létesítményt választják.

A probléma akkor merül fel, amikor egyes létesítmények alulhasználtak maradnak, és kevés felhasználó választja őket legközelebbi lehetőségként. Ezeket a létesítményeket áthelyezni költséges lehet, ezért a kutatás célja egy költséghatékonyabb megoldás kidolgozása: az utak fejlesztése, vagyis a közlekedési idő csökkentése bizonyos útszakaszokon annak érdekében, hogy az RNN mérete megnövekedjen.

Az **Expanding Reverse Nearest Neighbors (ERNN)** problémát azzal a céllal definiálták, hogy az adott hálózatban lévő létesítmények RNNméretét növelje anélkül, hogy azok fizikai helyzetét módosítani kellene. Ehelyett a megoldás az utak (élek) fejlesztésére összpontosít, ami csökkenti a közlekedési időt, és ezáltal több felhasználót vonz a cél létesítményhez.

Az ERNN-probléma azonban NP-nehéz és APX-nehéz, ami azt jelenti, hogy nehéz megtalálni az optimális megoldást hatékonyan. A dolgozat célja egy új megközelítés kidolgozása, amely hatékony algoritmusokat alkalmaz az ERNN-probléma gyakorlati megoldására.

Motiváció

A valós közlekedési hálózatokban egyes létesítmények, mint például tűzoltóállomások, kórházak vagy metróállomások, nem érik el a maximális kihasználtságukat, mivel távoli helyeken helyezkednek el. Ennek következtében kevés felhasználó választja ezeket legközelebbi lehetőségként, és így alulhasználtak maradnak. Ezzel szemben más létesítmények túlterheltek, mivel a felhasználók többsége ezeket választja.

Ez az egyenlőtlen kihasználtság pazarlást jelent, mivel az alulhasznált létesítmények nem teljesítik ki a potenciáljukat, miközben a forgalmas létesítmények terhelése nő. Például egy távoli tűzoltóállomás nem tudja kiszolgálni a környék összes felhasználóját, ha nincs megfelelő útvonal a hozzáféréshez. Ennek eredményeként a felhasználók inkább egy távolabbi, de könnyebben elérhető állomást választanak. A megoldás logikus lépése lehetne a létesítmény áthelyezése, de ez költséges és gyakran logisztikailag megoldhatatlan.

A dokumentum ezért egy másik, költséghatékony megközelítést kínál: az utak fejlesztését. Az utak szélesítése, gyorsforgalmi sávok építése, vagy az utak egyéb fejlesztései csökkenthetik a közlekedési időt, és így növelhetik a távoli létesítmények elérhetőségét, anélkül, hogy a létesítményeket fizikailag át kellene helyezni.

Kapcsolódó munkák

Számos kutatás foglalkozott már közlekedési hálózatok fejlesztésével, azonban a legtöbb munka elsősorban a hálózati átmérő minimalizálására vagy a késedelem csökkentésére összpontosított. Ezek a kutatások általában a globális hálózati jellemzők javítására irányultak, nem pedig az egyes létesítményekhez tartozó felhasználói csoportok (RNN) növelésére.

Az ERNN-probléma új megközelítést jelent a közlekedési hálózatok optimalizálásában, hiszen egy adott létesítmény RNN-méretének növelésére koncentrál, figyelembe véve az adott létesítmény kihasználtságát és az infrastruktúra javításának költségeit.

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.

A probléma formális definíciója

Az ERNN probléma formálisan a következőképpen definiálható: Adott egy gráf G(V,E,W), ahol a csomópontok (felhasználók és létesítmények), Eaz élek (utak), és az élek súlyai (utazási idők). 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.

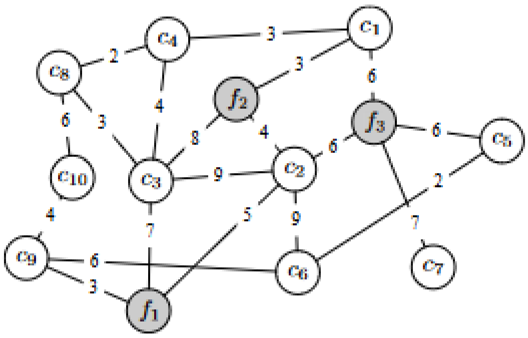


Figure 1: The Example Graph G

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 dolgozat egy 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 ← 0;
    dist(v) ← −1, visited(v) ← 0, for ∀v ∈ V;
    push f into Q;
    while Q is not empty do
        pop u from Q;
        visited(u) ← 1;
        for v ∈ N(u) and visitd(v) = 0 do
            if dist(u) + w(u, v) < dist(v) then
                dist(v) ← dist(u) + w(u, v);
                push v into Q;
            if e = (u, v) ∈ M and not used then
                dist((e), f) ← dist(u);
                Δ_e ← {c ∉ RNN_G(f) | dist_G(c, NN_G(c)) >
                  dist((e), f)};
                ub(e) ← |Δ(e) ∪ RNN_G(f)|;
                if ub(e) ≤ opt then stop this round;
                compute RNN_G'(f) in G' using Algorithm 1;
                if opt > |RNN_G'(f)| then
                    opt ← |RNN_G'(f)|, ans ← e;
    M ← M \ ans, A ← A ∪ ans;
    G ← upgrade ans in G, b ← b − 1;
return edges A;
```

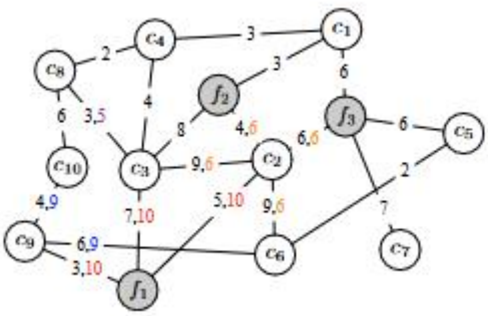
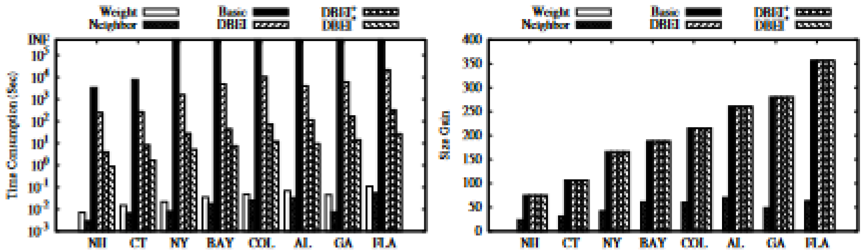


Figure 3: The Execution Process of Algorithm 3

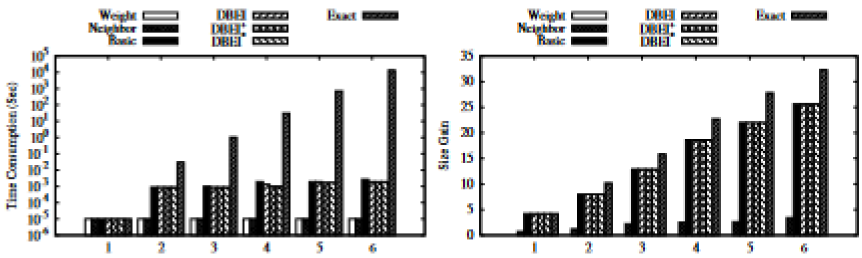
Table 1: Description of Datasets

Name	$\|V\|$	$\|E\|$
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

Az ERNN-probléma megoldásának számos gyakorlati alkalmazása van. A közlekedési rendszerek optimalizálása során a cél az, hogy a lehető legtöbb felhasználó számára biztosítsák a leggyorsabb hozzáférést a létesítményekhez. Ez magában foglalhatja a metróhálózatok optimalizálását, a tűzoltóságok elérhetőségének javítását, vagy az egészségügyi intézmények elérésének hatékonyabbá tételét.

Következtetések

A dolgozat 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.