УДК 681.518(045)

Гамаюн В. П., д. т. н.,
Мацуєва Х. А.

МОДЕЛЮВАННЯ МАРШРУТИЗАЦІЇ ГРАНЯМИ ГРАФАМИ ОДИНИЧНИХ КІЛ

Національний авіаційний університет
kafpi_itp@ukr.net
mchrissten@gmail.com

Розглянуто особливості застосування графів одиничних кіл у моделюванні маршрутизації гранями. Визначено основні геометричні властивості взаємодії вузлів у графах одиничних кіл. Запропоновано алгоритм обчислення шляху доставки пакету, який базується на застосуванні методу обчислення віртуального шляху у плоских віртуальних графах

Ключові слова: географічна маршрутизація; бездротові динамічні мережі; граф одиничних кіл; віртуальний граф; алгоритм маршрутизації

Вступ

У теорії графів графом одиничних кіл називається граф перетину сімейства одиничних кіл на евклідовій площині. Тобто, утворюється вершина для кожного кола і дві вершини з'єднуються ребром, якщо відповідні кола перетинаються [1].

Можливо кілька варіантів визначення графа одиничних кіл, еквівалентних з точністю до вибору коефіцієнта [1]:

– Граф перетину кіл або окружностей однакового радіусу.

– Граф, який сформований з набору кіл однакового радіусу, в якому два кола з'єднані ребром, якщо центр однієї окружності знаходиться всередині іншої.

– Граф, який утворений з набору точок на евклідовій площині, в якому дві точки з'єднані ребром, якщо відстань між цими точками менше деякого порогу.

Починаючи з роботи Хьюсона і Сена [2], графи одиничних кіл використовуються в інформатиці для моделювання топології бездротових динамічних мереж. У цьому випадку вершини з'єднані прямим бездротовим зв'язком без базової станції. Передбачається, що всі вершини однорідні і забезпечені всеспрямованими антенами. Розташування антени моделюється точками на евклідовій площині, а область, де сигнал може бути отриманий інший вершиною, моделюється колом. Якщо всі вершини мають передавачі однакової потужності, ці кола матимуть один і той же радіус. Випадкові геометричні графи, утворені як графи одиничних кіл з випадковими центрами, можна використовувати для моделювання фільтрації та деяких інших явищ [2].

Маршрутизація, що базується на місцезнаходженні, також відома як геометрична маршрутизація або географічна, була запропонована для вирішення проблеми масштабування. Замість використання топологічної інформації, протоколи даного виду маршрутизації використовують географічну інформацію про розташування вузлів для маршрутизації пакетів. Дані протоколи припускають, що вузли знають своє власне географічне положення (наприклад, від системи глобального позиціонування [3]) і вихідному вузлу відоме розташування вузла призначення.

Одним із видів географічної маршрутизації є маршрутизація гранями, яка гарантує доставку пакетів у статичних зв'язаних плоских графах [3, 5, 6]. Вона застосовується на плоскому графі, відповідно пакет передається по границях граней, які перетинаються відрізком, що з'єднує вихідний вузол і вузол призначення.

Геометричні властивості графів одиничних кіл

З метою визначення інформації, що необхідна на кожному вузлі під час до-

ставки пакету, визначимо властивості графів одиничних кіл. Наступна лема визначає геометричну властивість будь-яких двох ребер, які перетинають одне одного у графах одиничних кіл.

Визначення 1. Для будь-якого ребра *(u,v)*, лінзообразна область з хордою *(u,v)* є перетином двох кіл з одиничним радіусом, які містять *(u,v)* як хорду.

Затемнена область на рис. 1 є прикладом. Якщо вузол знаходиться у даній області з хордою *(u,v)*, то він є сусідом як *u* ,так *v*.

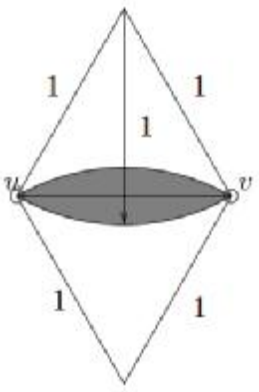

Рис. 1. Лінзообразна область з хордою *(u,v)*

Лема 2. Якщо у графах одиничних кіл ребро *(x, y)* перетинає ребро *(u,v)*, і ні *x*, ні *y* не знаходяться у лінзообразній області з хордою *(u,v)*, то *x* і *y* є сусідами *u*, або обидва сусіди *v*.

Доказ. У графах одиничних кіл, кожне ребро має довжину трохи більше 1. Розглянемо рис. 2., на якому дві прямі лінії паралельні *(u,v)* і мають відстань 1 від *(u,v)*, і два кола з центром у точці *u і v*, відповідно, з радіусом 1. Таким чином, товста крива є набором з усіх точок, які знаходяться на відстані 1 від *(u,v)*. Припустимо, що *(x.y)* перетинає *(u,v)* у точці *z*. Тоді відстань від x до *z* і від *x* до *(u,v)* є не більше 1; і відстань від *y* до *z*, і відповідно до *(u,v)* є також не більше *1*. Отже, обидва *x* і *y* розташовані всередині або на товстій кривій.

Далі, доведемо, що не дивлячись на припущення леми, *x* є сусідом принаймні одного з *u або v*. Припустимо, що це не так. Тоді *x* знаходиться в одній з двох трикутних областей, які не охоплюються двома колами, наприклад, одна з них сформована відрізком між *a* і *b*, дугою між *a* і *c*, і дугою між *b* і *c* (рис. 3).

Рис. 2. Доведення леми 2

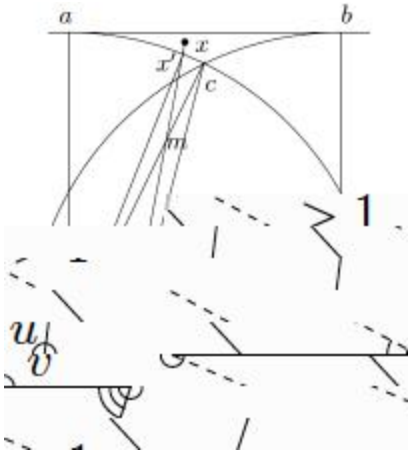

Рис. 3. Доведення леми 2

Припустимо, що *(x, y)* перетинає *(u, v)* і *y* знаходиться не у лінзообразній області з хордою *(u,v)*. Тоді *(x, y)* повинне перетинати нижню дугу між *u* і *v*, припустимо у точці *y'*. Крім того, *(x, y)* повинне перетинати дугу між *a* і *c* або дугу між *b* і *c*. Припустимо, *(x, y)* перетинає дугу між *a* і *c* у точці *x'*, і *(x, y)* перетинає відрізок між *c* і *u* у точці *m* (рис. 3). З нерівності трикутника, $|x'm| + |mu| \geq |ux'|$, і $|y'm| + |mc| \geq |cy'|$. Відомо, що $|ux'| = |cy'| = 1$. Таким чином, $|x'm| + |mu| + |y'm| + |mc| = |x'y'| + |uc| \geq 2$. З $|uc| = 1$, випливає, що $|x'y'| \geq 1$. Оскільки *x* не є сусідом *u*, $|xu| > 1$, то $|xx'| > 0$. Оскільки *y* не у лінзообразній області з хордою *(u,v)*, $|yy'| > 0$. Отже, $|xy| = |x'y'| + |xx'| + |y'y| > |x'y'| \geq 1$.. Це суперечить тому,

лом u, тому що він має інформацію про два сусідні переходи. Таким чином, вузол u перевіряє всі можливі ребра перетину (u, v) і знаходить точку перетину β, що знаходиться найближче до α, ребро (w, x), яке містить наступне віртуальне ребро після (β, α) за годинниковою стрілкою навколо β, і шлях π від u до w.

Шлях π, що повертається алгоритмом EPND, має не більше двох переходів, тому що обидві кінцеві точки ребра, що перетинає (u, v), знаходяться на відстані не більше двох переходів від u, відповідно до слідства 3. Якщо немає ніяких ребер, що перетинають (u, v), то початковим вузлом ребра, яке містить наступне віртуальне ребро є v, що є сусідом u. Таким чином, у 2HOP, шлях π, який повертається алгоритмом EPND, не зберігається у заголовку пакета. Вузол u просто перевіряє, чи є початковий вузол ребра *packet.next_edge* його сусідом. Якщо це так, то він пересилає пакет безпосередньо до цього вузла; в іншому випадку, він пересилає пакет у вузол, що є сусідом для них обох. Коли цей проміжний вузол приймає пакет, він визначає, що він не є початковим вузлом *packet.next_edge*. У цьому випадку, він просто передає пакет на визначений початковий вузол.

Розглянемо процедуру перемикання (переходу) між гранями. Перехід на наступну грань відбувається, коли пакет досягає точки на границі поточної віртуальної грані, яка знаходиться ближче до місця призначення, ніж поточна відправна точка. Таким чином, після того, як вузол обчислив кінцеву точку віртуального ребра, що міститься у *packet.next_edge*, він перевіряє, чи є відстань від будь-якої точки на цьому віртуальному ребрі до вузла призначення менше, ніж відстань від поточної початкової точки. Якщо це так, найближча точка до місця призначення на цьому віртуальному ребрі буде новою відправною точкою для наступної віртуальної грані. Вузол викликає INIT-2HOP з метою знаходження ребра, що містить перше віртуальне ребро вздовж границі наступної віртуальної грані і оновлює поля заголовку пакета. Потім пакет повинен бути перенаправлений у початковий вузол нового *packet.next_edge*, щоб почати обхід нової віртуальної грані. Цілком можливо, що початковий вузол є поточним. У цьому випадку, передача пакетів не потрібна і поточний вузол просто виконує алгоритм 2HOP знову. Якщо початковий вузол нового *packet.next_edge* не є поточним, то він знаходиться на відстані не більше двох переходів, тому що обидві кінцеві точки нового *packet.next_edge* знаходяться в межах двох переходів від поточного вузла. Таким чином, поточний вузол пересилає пакет до цього вузла способом, який описано вище.

Повний код алгоритму 2HOP наведено на рис. 6.

Теорема 7. 2HOP моделює маршрутизацію гранями у віртуальному плоскому графі, який складається з зв'язаних графів одиничного кола.

Алгоритм 2HOP
* Виконується вузлом u (вузол, який в даний час має пакет)
* Інформація, що зберігається у заголовку пакету є:
• *packet.destination*, вузол-адресат пакету;
• *packet.next_edge*, мережеве ребро, яке містить наступне віртуальне ребро;
• *packet.last_point*, кінцева точка попереднього віртуального ребра або початкова точка на поточній віртуальній грані, якщо *packet.next_edge* містить перше віртуальне ребро на даній грані;
• *packet.distance*, відстань від початкової точки поточної віртуальної грані до вузла призначення.

begin
if u is *packet.destination* **then**
release the packet; return
if *packet.destination* is a neighbor of u **then**
forward the packet to *packet.destination*; return
if u is not the beginning node of *packet.next_edge* **then**
forward the packet to the beginning node of *packet.next_edge*; return
* u є початковим вузлом *packet.next edge*
let v denote the ending node of *packet.next edge*
EPND((u, v), *packet.last point*, β, (w, x), π)

```
        d ← shortest distance from a point on edge
(packet.last point, β) to       packet.destination
        if d < packet.distance  then   *
перемикання граней
          p ← the closest point to packet.destination
on edge (packet.last point, β)
          INIT-2HOP(p)
             if u is the beginning node of packet.next
edge  then
                  2HOP
                  return
        else
                  packet.next edge ← (w, x)
                  packet.last point ← β
        * переміщення пакету до початкового
вузла packet.next_edge
        let y denote the beginning node of
packet.next_edge
        if y is a neighbor of u  then
          forward the packet to y
        else
           find node z such that z is a neighbor of
        both u and y
           * z є v або кінцевим вузлом
        packet.next_edge
          forward the packet to z
        return
end
```

Рис. 6. Алгоритм 2HOP

Висновок

У результаті проведених досліджень показано, що визначення основних геометричних властивостей взаємодії вузлів у графах одиничних кіл дозволяє змоделювати маршрутизацію гранями у графах, які не містять основного зв'язаного плоского підграфу. Тобто можливе застосування маршрутизації гранями безпосередньо на не планарний граф мережі, без отримання плоского підграфу. Було розроблено протокол маршрутизації у графах одиничних кіл, який ілюструє метод обчислення віртуального шляху у плоских віртуальних графах [7]. Відповідно до даного протоколу, було проаналізовано загальний процес маршрутизації гранями методом віртуальних графів у графах одиничних кіл і запропоновано алгоритми обчислення шляху доставки пакету.

Список літератури

1. Brent N. Clark, Charles J. Colbourn, David S. Johnson Unit disk graphs // Discrete Mathematics. – 1990. – В. 1-3. – Т. 86. – С. 165–177.

2. Mark L. Huson, Arunabha Sen Military Communications Conference, IEEE MILCOM '95. – 1995. – Т. 2. – С. 647-651.

3. F. Kuhn, R. Wattenhofer, A. Zollinger. Asymptotically optimal geometric mobile ad-hoc routing // In DIALM '02: Proceedings of the 6th international workshop on Discrete algorithms and methods for mobile computing and communications. – New York, ACM Press, 2002. – P. 24-33.

4. E. Kranakis, H. Singh, J. Urrutia. Compass routing on geometric networks // In Proc. 11th Canadian Conference on Computational Geometry. - Vancouver, August 1999. – P. 51-54.

5. P. Bose, P. Morin, I. Stojmenovic, J. Urrutia. Routing with guaranteed delivery in ad hoc wireless networks // Wireless Networks, №7(6). – 2001. – P. 609–616.

6. B. Karp, H. T. Kung. GPSR: greedy perimeter stateless routing for wireless networks // In MobiCom '00: Proceedings of the 6th annual international conference on Mobile computing and networking. – 2000. – P. 243–254.

7. Гамаюн В.П., Мацуєва Х.А. Моделювання маршрутизації гранями у бездротових динамічних мережах за допомогою віртуальних графів // Проблеми інформатизації та управління, №2(46). – К: Видавництво НАУ, 2014. – С. 14–19.

Статтю подано до редакції 07.09.2015