

16. Realizacja zadań w grafie przez grupę mobilnych jednostek

Tasks realization on graphs by a group of mobile agents

Dorota Urbańska-Osula⁽¹⁾⁽²⁾

⁽¹⁾ Katedra Algorytmów i Modelowania Systemów, Politechnika Gdańska

⁽²⁾ Badania częściowo sponsorowane przez Narodowe Centrum Nauki, Polska, grant numer 2015/17/B/ST6/01887

Opiekun naukowy: dr hab. inż. Dariusz Dereniowski

Dorota Urbańska-Osula: dorurban@student.eti.pg.gda.pl

Słowa Kluczowe: eksploracja, przeszukiwanie, agenci, optymalizacja

Grupa mobilnych jednostek, nazywanych także agentami, jest umiejscowiona w jednym lub wielu wierzchołkach grafu nazywanych bazami. Stamtąd poruszając się po z góry znanym (offline) lub nieznanym (online) grafie muszą wykonać powierzone im zadanie, takie jak przeszukiwanie grafu, spotkanie, dekontaminacja grafu czy wybór lidera. Celem jest znalezienie optymalnej, rozproszonej, deterministycznej strategii (sekwencji ruchów jednostek), która umożliwi realizację zadania. Poniższy artykuł jest artykułem przeglądowym skupiającym się na najbardziej aktualnych, wybranych modelach i problemach.

1. Wstęp

Przetwarzanie rozproszone odgrywa coraz większą rolę w wielu dziedzinach Informatyki. Dzięki szybkiemu rozwojowi techniki mamy możliwość wyposażać poszczególne jednostki systemu w coraz więcej inteligencji i samodzielności, co w końcowym efekcie prowadzi do całkowitej rezygnacji z centralnego zarządzania. Grupy zaprogramowanych jednostek w bardzo dobry sposób odzwierciedlają zjawiska obserwowane w naturze, takie jak rój pszczół, kolonia mrówek czy nawet grupa ludzi. Odpowiednio zaprogramowane symulują współpracę grupową do osiągnięcia wyznaczonego im celu. W poniższym artykule autorzy opisują rodzaje problemów jakie mogą zostać postawione grupie niezależnych mobilnych jednostek na *grafie* (definicja grafu została podana w Rozdz. 2). Oprócz literatury podstawowej zostały również zaprezentowane wybrane, aktualne modele i zagadnienia. Ze względu na obszerność prezentowanego zagadnienia autorzy musieli ograniczyć się do subiektywnie wybranych najważniejszych i/lub najciekawszych prac z danej tematyki. W Rozdziale 2 znajduje się opis zagadnienia oraz podane są wszystkie niezbędne definicje. Rozdział 3 podaje opis szczegółowy oraz odniesienia do przekroju literatury dla wybranych pięciu problemów. W kolejnej części pracy znajdują się wnioski (Rozdział 4) oraz spis literatury w porządku alfabetycznym (Rozdział 5).

2. Opis zagadnienia

Grafem nazywamy parę $G = (V, E)$, gdzie V jest niepustym zbiorem skończonym, a E jest dowolnym podzbiorem zbioru nieuporządkowanych par elementów zbioru V . Elementy zbioru V nazywamy *wierzchołkami* (lub *węzłami*) i ich liczbę oznaczamy będziemy przez $n = |V|$, zaś zbiór E jest zbiorem *krawędzi*, którego moc oznaczamy przez $m = |E|$. Wyróżniamy wiele klas grafów, w tym najpopularniejsze to: *ścieżka* P_n , *cykl* C_n , *graf pełny* K_n oraz *drzewo* T (gdzie indeksy dolne oznaczają liczbę wierzchołków grafu). Na Rysunku nr 1 zostały zaprezentowane przykładowe grafy z wymienionych wyżej klas.

Poniższe rozważania ograniczamy do grafów spójnych (wszystkie wierzchołki grafu tworzą jeden komponent) oraz nieskierowanych (krawędzie nie mają nadanego kierunku).

Grupa mobilnych jednostek zwanych agentami, robotami lub też poszukiwaczami (z ang. searcher) zostaje umiejscowiona w jednym (lub więcej) wierzchołkach grafu zwanymi bazami. Jeśli znają one budowę całego grafu mówimy o wersji offline, a jeśli jedynymi informacjami jakie

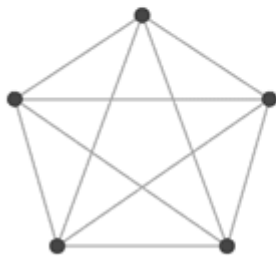
dysponują jest ich aktualne położenie - o wersji online. Dopuszczamy komunikację między agentami: najpopularniejszym modelem jest założenie, że wierzchołki mają pamięć lokalną z której agenci mogą korzystać w okupowanym aktualnie węźle. Przed agentami zostaje postawione do zrealizowania jedno z zadań opisanych szczegółowo w rozdziale 3: rendez-vous, wybór spośród siebie lidera, dekontaminacja grafu, eksploracja grafu czy lokalizacja wrogiego węzła. Celem jest znalezienie strategii, tj. sekwencji ruchów agentów, która będzie minimalizowała liczbę użytych robotów, czas w jakim zostanie wykonane zadanie lub wielkość baterii w jaką muszą zostać wyposażeni. Za ruch agenta uważane jest zostanie w tym samym miejscu lub przesunięcie się po przystającej krawędzi do sąsiadującego wierzchołka. W tym artykule rozważane są strategie rozproszone (nie ma centralnego dowodzenia, a każdy z agentów podejmuje własne decyzje na podstawie swoich obserwacji oraz informacji zebranych od pozostałych agentów) oraz deterministyczne (brak losowości w wyborach agentów). Ponadto jeśli nie jest napisane inaczej rozważany jest synchroniczny model czasu, tj. jeden ruch agenta trwa jedną jednostkę czasu.



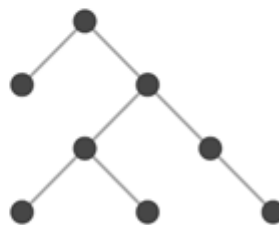
Rys.1a. Ścieżka P_6 .



Rys.1b. Cykl C_5 .



Rys.1c. Graf pełny K_5 .



Rys.1d. Przykładowe drzewo T o 8 wierzchołkach.

Rys.1. Przykładowe klasy grafów.

3. Przegląd literatury

Rendez-vous, zwany potocznie w literaturze jako problem matki z zagubionym dzieckiem w supermarkecie. Dwóch agentów zostaje umieszczonych w dwóch różnych wierzchołkach grafu i celem jest okupowanie tego samego wierzchołka w tym samym czasie, innymi słowy: spotkanie. Problem, chociaż może wydawać się prosty, w rzeczywistości jest problemem należącym do klasy problemów NP-trudnych, tj. nie jest znane rozwiązanie tego problemu w czasie wielomianowym. Co więcej, bez żadnych dodatkowych założeń np. o rozpoznawaniu kierunków poruszania się, problem w wersji online jest NP-trudny nawet dla cyklu – innymi słowy matka z dzieckiem może biegać w kółko po okręgu nigdy się nie spotykając. Problem ten jest szeroko znany, więcej informacji o jego wersji podstawowej znajdziemy w książce (Alpern and Gal 2003). Fraigniaud i Pelc w 2008 (Fraigniaud and Pelc 2008) po raz pierwszy dokonali analizy zachowania dwóch identycznych agentów bez możliwości zapisywania niczego w wierzchołkach dla drzew. Farrugia i in. (Farrugia et al. 2015) rozważyli model dla cyklu, ale gdy agenci nie startują razem, lecz moment startu dla drugiego agenta jest wybierany przez wrogiego przeciwnika (jest najgorszy z możliwych dla



zadanego algorytmu). Ostatnim modelem o którym tu wspomniemy jest model zaproponowany przez Dereniowskiego i in. (Dereniowski et al. 2014) w którym to agenci mają różne prędkości poruszania się po różnych krawędziach. Każdy z agentów zna całą topologię sieci, wzajemne pozycje startowe i własne prędkości poruszania się, lecz prędkości poruszania się drugiego agenta pozostają tajemnicą. Rozważany jest optymalny czas spotkania oraz ilość potrzebnych do wymiany informacji między agentami, aby wynik optymalny mógł zostać otrzymany.

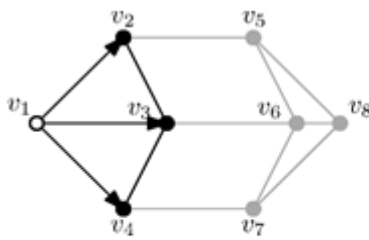
Wybór lidera - często zanim agenci przystąpią do realizacji innego zadania, muszą wyróżnić spomiędzy siebie jednego z nich, tzw. lidera. Innymi słowy po zadanym czasie oczekujemy, że każdy z agentów będzie wiedzieć czy jest liderem czy nie. Do tego problemu zakładamy, że agenci są całkowicie nierozróżnialni (nie posiadają unikatowych ID), gdyż w przeciwnym razie rozwiązanie mogłoby być banalne, tj. wybierz agenta o najmniejszym ID. Jest to problem, ze względu na swoje szerokie zastosowanie w teorii przetwarzania rozproszonego, studiowany od wielu lat (np. praca Itkisa i in. (Itkis et al. 1995) dotycząca zachowania agentów na cyklach czy też analiza czasu wykonania zadania oraz rozmiarów przesyłanych między sobą wiadomości dla grafów pełnych wykonana w roku 2005 przez Villadangos i in. (Villadangos et al. 2005)). Jednym z nowych, ciekawych wyników został zawarty w pracy Dereniowskiego i Pelca z roku 2014 (Dereniowski and Pelc 2014). Autorzy przedstawiają pełną charakterystykę wszystkich możliwych konfiguracji początkowych agentów dla dowolnych grafów w asynchronicznym środowisku dla których realizacja tego zadania jest możliwa. Dodatkowym utrudnieniem jest, że agenci poruszają się po nieznanym grafie w którym krawędzie wychodzące z jednego wężła są nierozróżnialne (tj. agent nie rozróżnia kierunków).

Dekontaminacja grafu oznacza jego oczyszczenie. Jest to model tzw. „trującego gazu w labiryncie” lub „wszechwiedzącego, superszybkiego uciekiniera”. Agenci startują z jednego wierzchołka i muszą oczyścić cały graf, który na początku jest zanieczyszczony. Węzły zostają oczyszczone gdy zostają odwiedzone przez agenta. Rozróżniamy trzy sposoby czyszczenia krawędzi: edge search (krawędź grafu zostaje oczyszczona gdy agent się po niej prześliznie), node search (krawędź grafu zostaje oczyszczona gdy w tym samym momencie obydwie jej końce są okupowane przez agentów) oraz mixed search (połączone edge i node search). Bienstock i Seymour w 1991 roku udowodnili, że wybór modelu nie wpływa istotnie na wynik, gdy celem jest minimalizacja liczby agentów (Bienstock and Seymour 1991). Ważnym pojęciem jest rekontaminacja wierzchołka, która ma miejsce gdy jeden z wierzchołków został już oczyszczony, ale nie jest okupowany już przez agenta i jednocześnie jeden z jego sąsiadów jest skażony. Wtedy „trujący gaz” może wedrzeć się do oczyszczonej już części, co ilustruje Rysunek nr 2.

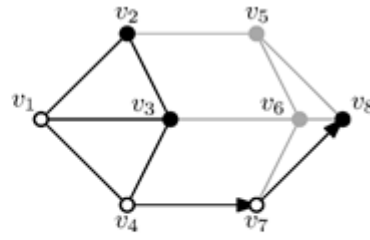
Zakładamy, że strategie, których szukamy są strategiami monotonicznymi, tzn. nie dopuszczona jest w nich rekontaminacja. Problem znalezienia minimalnej liczby agentów, która wykona to zadanie jest NP-trudny już dla drzew ważonych (każdy wierzchołek ma przypisaną liczbę całkowitą, która oznacza liczbę agentów jaka się musi na nim znaleźć by został oczyszczony), co zostało pokazane przez Dereniowskiego (Dereniowski 2010). Wyniki dla pozostałych klas grafów można znaleźć w: drzewa nieważone (Barrière et al. 2002), d-wymiarowa hiperkostka (Flocchini et al. 2008), pierścien i torus (Flocchini et al. 2007) oraz graf Sierpińskiego (Luccio 2008).

Celem eksploracji grafu jest odwiedzenie przez agentów wszystkich jego wierzchołków. W przeciwieństwie do dekontaminacji, nie wymagamy tutaj monotoniczności, ale za to zawsze graf jest agentem całkowicie nieznanym. Czytelnik może myśleć o tym modelu jako o grupie robotów, które zostały postawione w jednym punkcie terenu i ich celem jest zbudowanie jego mapy. Szukana strategia może być (tak jak w przypadku dekontaminacji) optymalna pod względem liczby agentów, które to podejście możemy znaleźć w pracy Das i in. (Das et al. 2015). Rozważane tam są roboty o ograniczonej wielości baterii, która pozwala im podróżować jedynie na określony dystans (gdyby tak nie było, to oczywiście problem staje się trywialny gdyż jeden robot jest wystarczający do eksploracji każdego grafu za pomocą chociażby algorytmu DFS). Kolejnym podejściem jest minimalizacja energii robotów, tj. maksymalnego dystansu przebytego przez każdego z nich. W pracy z 2006 Dynia i in. (Dynia et al. 2006) prezentują ograniczenie dolne czasu przy znanej liczbie robotów dla drzew. Jednakże najpopularniejszym podejściem jest minimalizacja czasu eksploracji

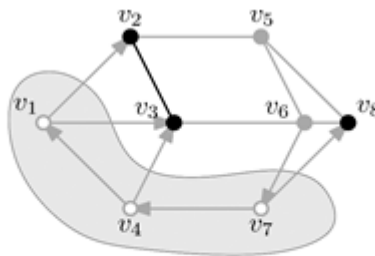
przy zadanej wielkości grupy robotów, które to możemy znaleźć w m.in. (Fraigniaud et al. 2006; Brass et al. 2011; Dereniowski et al. 2015).



Rys.2a. Trzech-agentów startuje z wierzchołka v_1 .



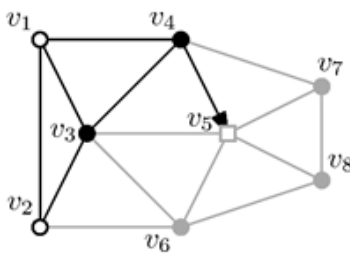
Rys.2b. Jeden z-agentów idzie do wierzchołka v_8 oczyszczając po drodze v_7 .



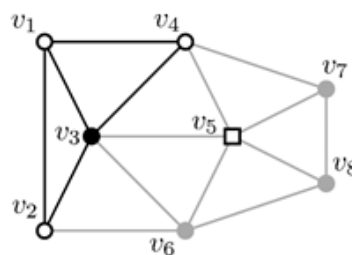
Rys.2c. Rekontaminacja następuje najpierw w niestrzeżonym v_7 , a następnie w v_4 oraz v_1 .

Rys.2. Przykładowa rekontaminacja części grafu; czarne pełne kropki oznaczają okupowane aktualnie wierzchołki; czarne puste okręgi oznaczają już odwiedzone węzły; szare pełne kropki oznaczają wierzchołki nieodwiedzone; szare puste okręgi to wierzchołki zrekontaminowane; czarne krawędzie oznaczają wyczyszczone krawędzie, zaś szare te skażone; czarna strzałka oznacza ruch który w następnej kolejności zostanie wykonany przez agenta, zaś szara kierunek rekontaminacji.

Lokalizację wrogiego węzła dzielimy na problem black hole (BH) oraz black virus (BV). BH jest procesem stacjonarnym, tj. nie może się poruszać, rezydującym w węzłach grafu (Zarrad and Daadaa, 2013). Gdy agent natrafi na zakażony węzeł – umiera. Celem drużyny jest zlokalizowanie położenia BH w grafie (Rys. 3).



Rys.3a. Agent okupujący v_4 wykonuje ruch do v_5 , **Rys.3b.** Agent umiera, zaś agent okupujący wierzchołek v_3 zna już położenie BH, co kończy realizację zadania.



Rys.3. Lokalizacja BH w przykładowym grafie; czarne pełne kropki oznaczają okupowane aktualnie wierzchołki; czarne puste okręgi oznaczają już odwiedzone wierzchołki; szare pełne kropki oznaczają wierzchołki nieodwiedzone; szary kwadrat to wierzchołek zawierający BH; czarne krawędzie oznaczają odwiedzone krawędzie, zaś szare te nieodwiedzone; czarna strzałka oznacza ruch który w następnej kolejności zostanie wykonany przez agenta.

4. Podsumowanie

Modele zaprezentowane w tym artykule mają zastosowanie do wielu rzeczywistych problemów, takich jak: organizacja realizacji zadania w przetwarzaniu rozproszonym na wielu maszynach, oczyszczenie sieci komputerowej z wirusa, lokalizacja wrogiego węzła w sieci czy też budowa mapy trudnodostępnego terenu jak np. powierzchni innej planety. Tematyka badania złożoności obliczeniowej algorytmów rozproszonych na grafie jest dziedziną bardzo dynamicznie rozwijającą się, a każdy rozwiązany problem prowadzi do narodzin kilku nowych. Wymienione powyżej pięć zagadnień pozostaje ciągle nierozwiązane dla wielu klas grafów i różnych parametrów modeli, takich jak liczba i konfiguracja pozycji startowych czy widoczność lub sposób komunikacji agentów. Co więcej, ciągle dla wielu sytuacji jest nieudowodniona NP-zupełność problemu (czyli jego nierozwiązywalność w realnym, skończonym czasie). Często też, jak np. w dekontaminacji czy lokalizacji wrogiego węzła, większość powstałych prac porusza szukanie optymalnych strategii pod kątem minimalizacji liczby agentów, zostawiając tym samym szeroką lukę do wypełnienia dla badań czasu realizacji powierzonych zadań.

5. Literatura (wymieniona alfabetycznie)

- Alpern S, Gal S (2003) *The Theory of Search Games and Rendezvous*. Kluwer.
- Barrière L, Flocchini P, Fraigniaud P et al. (2002) Capture of an intruder by mobile agents. In Proc. SPAA 200-209.
- Bienstock D, Seymour P (1991) Monotonicity in graph searching. *J. Algorithms* 12(2): 239-245.
- Brass P, Cabrera-Mora F, Gasparri A et al. (2011) Multirobot Tree and Graph Exploration. *IEEE Trans. Robotics* 27(4) 707-717.
- Cai J, Flocchini P, Santoro N (2014) Decontaminating a Network from a Black Virus. *IJNC* 4(1):151-173.
- Das S, Dereniowski D, Karousatou C (2014) *Collaborative Exploration by Energy-Constrained Mobile Robots*. Springer International Publishing, 2014 p. 357-369.
- Dereniowski D (2010) Connected searching of weighted trees. Springer Berlin Heidelberg, 2010 p. 330-341.
- Dereniowski D, Disser Y, Kosowski A et al. (2015) Fast collaborative graph exploration. *Inf. Comput.* 243: 37-49.
- Dereniowski D, Klasing R, Kosowski A et al. (2014) Rendezvous of Heterogeneous Mobile Agents in Edge-Weighted Networks. *Theoretical Computer Science* 608: 219-230.
- Dereniowski D, Pelc A (2014) Leader election for anonymous asynchronous agents in arbitrary networks. *Distributed Computing* 27(1): 21-38.
- Dynia M, Korzeniowski M, Schindelbauer C (2006) Power-Aware Collective Tree Exploration. In Proc. Architecture of Computing Systems ARCS 2006 341-351.
- Farrugia A, Gasieniec L, Kuszner L et al. (2015) Deterministic Rendezvous in Restricted Graphs. In Proc. SOFSEM 2015 189-200.
- Flocchini P, Huang MJ, Luccio FL (2007) Decontaminating Chordal Rings and Tori Using Mobile Agents. *Int. J. Found. Comput. Sci.* 18(3): 547-563.
- Flocchini P, Huang MJ, Luccio FL (2008) Decontamination of hypercubes by mobile agents. *Wiley-Interscience* 52(3): 167-178.
- Fraigniaud P, Gasieniec L, Kowalski DR et al. (2006) Collective tree exploration. *Networks* 48(3): 166-177.
- Fraigniaud P, Pelc A (2008) Deterministic Rendezvous in Trees with Little Memory. In Proc. Distributed Computing, 22nd International Symposium, DISC 2008 242-256.
- Itkis G, Lin C, Simon J (1995) Deterministic, constant space, self-stabilizing leader election on uniform rings. In Proc. 9th Workshop on Distributed Algorithms 972: 288-302.
- Luccio FL (2008) Contiguous Search Problem in Sierpiński Graphs. *Theory of Computing Systems* 44(2): 186-204.

- Villadangos JE, Alberto Córdoba A, Federico Fariña F et al. (2005) Efficient Leader Election in Complete Networks. In Proc. 13th Euromicro Workshop on Parallel, Distributed and Network-Based 136-143.
- Zarrad A, Daadaa Y (2013) A Review of Computation Solutions by Mobile Agents in an Unsafe Environment. International Journal of Advanced Computer Science and Applications 4(4).