



Gdańsk, 14-16 maja 2012

ANALIZA ALGORYTMÓW STEROWANIA TOPOLOGIĄ SIECI SENSOROWYCH POD KĄTEM REALIZACJI DEDYKOWANEGO ROZWIĄZANIA SPRZĘTOWEGO

Streszczenie: W niniejszym artykule scharakteryzowano wybrane algorytmy samoorganizacji mobilnych sieci sensorowych, a także przeanalizowano je pod kątem zastosowania w projektowanym rozwiązaniu sprzętowym. W tym celu przedstawiono wyniki symulacji przeprowadzonych z wykorzystaniem zaimplementowanego oprogramowania. Podstawowymi czynnikami wpływającymi na wybór mechanizmu sterowania topologią sieci były niewielkie możliwości obliczeniowe węzła oraz ograniczona pojemność jego baterii.

1. WSTĘP

Sieci sensorowe są stosunkowo nową gałęzią rozwoju sieci bezprzewodowych. Ewolują jednak bardzo szybko, co jest wynikiem mnogości zastosowań w różnych dziedzinach, zarówno komercyjnych, jak i militarnych oraz przemysłowych. Podstawowym założeniem bezprzewodowych sieci sensorowych (*Wireless Sensor Networks* – WSN) jest zdecentralizowana akwizycja danych, a następnie ich cyfrowe przetwarzanie. Sieci sensorowe składają się z wielu inteligentnych urządzeń, zwanych węzłami, rozmieszczonych na pewnym obszarze, które oprócz wykrywania (bądź monitorowania) pewnego zjawiska przetwarzają uzyskane dane, a także posiadają własną pamięć i realizują komunikację z innymi węzłami sieci. Brak infrastruktury w sieciach sensorowych sprawia, że węzły muszą samodzielnie dokonywać konfiguracji sieci, a także zarządzać nią i kontrolować jej działanie. Z tego powodu każdy węzeł musi działać zarówno jako terminal oraz przekaźnik danych. Niestety zarządzanie siecią za pośrednictwem węzłów znacznie obciąża ich zasoby, co istotnie komplikuje projektowanie protokołów i optymalizację wydajności sieci sensorowych.

Głównym celem projektowania nowych rozwiązań dla sieci sensorowych jest efektywność energetyczna. W większości zastosowań bowiem sieci sensorowe cechują się ograniczonymi możliwościami wymiany i doładowania baterii w poszczególnych węzłach, w związku z czym należy starannie i efektywnie dysponować ich zasobami energetycznymi. Wymaga to zastosowania odpowiednich algorytmów sterowania topologią sieci, które należy uprzednio przeanalizować pod kątem zastosowań danej sieci i możliwości obliczeniowych wykorzystanych w niej modułów.

2. STEROWANIE TOPOLOGIĄ SIECI

Dynamiczne sterowanie topologią (*Topology Control*) jest jednym z elementów realnie wpływających na

wydajność pracy sieci i pozwalającym w istotny sposób zmniejszyć zarówno zużycie energii jak i poziom generowanych interferencji od poszczególnych jej elementów. Do algorytmów samoorganizacji sieci zalicza się jedynie rozwiązania pracujące podczas działania samej sieci i wykonywane przez sieć autonomicznie.

Efektywne wykorzystanie ograniczonych zasobów energetycznych sieci jest jednym z głównych zadań stojących przed projektantem sieci. W przypadku sieci sensorowych węzły zużywają istotną ilość energii na transmisję wiadomości, stąd redukcja energii wykorzystywanej przez łączność radiową jest elementem niezwykle ważnym.

Podstawowy podział metod sterowania topologią sieci pozwala wyróżnić mechanizmy homogeniczne i niehomogeniczne [1]. W pierwszym przypadku wszystkie węzły sieci transmitują z tą samą ustaloną mocą. Wadą tego rozwiązania jest jednak duże zużycie energii oraz wysoki poziom interferencji. Z tego powodu algorytmy homogeniczne nie są odpowiednie do zastosowania w sieciach sensorowych i nie będą dalej rozpatrywane. Zdecydowanie korzystniejsze, w kontekście sieci WSN, jest sterowanie niehomogeniczne, charakteryzujące się tym, że poszczególne węzły mogą wybierać różne moce transmisji poniżej pewnego maksymalnego poziomu nominalnego. Niehomogeniczne sterowanie topologią sieci można podzielić na trzy podgrupy: oparte na pozycji, kierunku oraz sąsiedzie, w zależności od tego jaki rodzaj informacji wykorzystywany jest do obliczania topologii. Zostały one opisane w kolejnych podrozdziałach.

2.1. Sterowanie oparte na pozycji

W mechanizmach sterowania opartych na pozycji (*Location Based*) konieczne jest posiadanie jak najdokładniejszej informacji o położeniu każdego węzła. Te informacje są później wymieniane pomiędzy węzłami i wykorzystywane do obliczenia topologii w pełni rozproszony sposób. Zakłada się, że wszystkie węzły (albo przynajmniej ich znacząca część) wyposażone są w odbiorniki GPS (*Global Positioning System*). Takie rozwiązania stosuje się zwykle w terenie otwartym z powodu trudnego odbioru sygnału GPS w budynkach.

Przykładem algorytmu opartego na pozycji jest LMST (*Local Minimum Spanning Tree*) [2]. Działanie algorytmu LMST polega na generowaniu przez każdy węzeł (na podstawie ustalonej listy sąsiadów) minimalnego drzewa opinającego (*Minimum Spanning Tree* – MST) według klasycznego algorytmu Prima na podstawie odległości euklidesowych. Następnie dokonywana

jest superpozycja lokalnych drzew opinających, w wyniku czego otrzymuje się ostateczną topologię sieci.

2.2. Sterowanie oparte na kierunku

Przy sterowaniu topologią opartym na kierunku (*Direction Based*) zakłada się, że poszczególne węzły nie znają swojej pozycji, ale mogą określić kierunek względem najbliższych sąsiadów i na tej podstawie dokonywane są obliczenia. Do określania kierunku AoA (*Angle of Arrival*), z którego nadaje węzeł, najczęściej wykorzystuje się dodatkową antenę kierunkową. Przewagą rozwiązań wykorzystujących AoA w stosunku do rozwiązań opartych na pozycji jest to, że mogą być one wykorzystywane zarówno w terenie otwartym jak i wewnątrz budynków.

Przykładowymi algorytmami opartymi na kierunku są CBTC (*Cone-based Topology Control*) [3] oraz DistRNG (*Distributed Relative Neighborhood Graph*) [4]. Algorytm CBTC tworzy ścieżki tak, aby w każdym wycinku koła o zadanym kącie znajdował się co najmniej jeden sąsiad. Natomiast w DistRNG ustanowienie połączenia pomiędzy dwoma węzłami u i v jest możliwe w przypadku, gdy w obszarze ograniczonym fragmentami okręgów o promieniach równych odległości pomiędzy węzłami u i v oraz środkach w tychże węzłach nie występuje żaden inny węzeł. Algorytmy te posiadają kilka istotnych zalet: są w pełni rozproszone, a także zachowują połączeniowość i wymagają jedynie informacji o kierunku nadejścia sygnału.

2.3. Sterowanie oparte na sąsiedzie

Algorytmy sterowania oparte na sąsiedzie (*Neighbour Based*) bazują na zdolności określenia liczby oraz tożsamości węzłów sąsiednich w zasięgu transmisji danego węzła. Z wykorzystaniem zestawu informacji o węzłach sąsiednich tworzona jest topologia na podstawie takich parametrów jak odległość, bądź jakość łącza.

Przykładowymi algorytmami opartymi na sąsiedzie są KNeigh (*k-Neighbours*) [5], LINT (*Local Information No Topology*) oraz LILT (*Local Information Link state Topology*) [6]. Idea działania algorytmu KNeigh jest bardzo prosta. Każdy węzeł rozgłasza swoją tożsamość z maksymalną mocą. Następnie po otrzymaniu informacji o tożsamości swoich sąsiadów zapamiętuje ich dane, przypisując każdemu odległość estymowaną na podstawie: pomiaru siły sygnału odebranego RSSI (*Received Signal Strength Indication*), czasu dostarczenia wiadomości TOA (*Time Of Arrival*) lub współczynnika jakości łącza LQI (*Link Quality Indicator*). Na bazie tych informacji każdy węzeł oblicza swoją listę k -najbliższych sąsiadów.

LINT natomiast dąży do zachowania połączenia z pewną liczbą węzłów sąsiednich zawartą w określonych granicach zdefiniowanych wcześniej jako pożądana liczba sąsiadów. W przypadku, gdy liczba sąsiadów spadnie poniżej wartości granicznej moc transmisji węzła jest zwiększana, jeśli natomiast węzeł ma w zasięgu zbyt dużą liczbę sąsiadów moc zostaje zmniejszona.

Algorytmy oparte na sąsiedzie posiadają kilka pożądanых cech: są stosunkowo proste, wymagają niewiele informacji, wymiana wiadomości pomiędzy węzłami sieci jest minimalna, a sam algorytm jest w pełni rozproszony i asynchroniczny. Posiadają jednak pewną wadę, a

mianowicie nie zachowują połączeniowości w najgorszym przypadku [5]. Oznacza to, że otrzymana topologia nie zawsze zachowuje taką samą połączeniowość co sieć transmitująca z maksymalną mocą.

3. OPROGRAMOWANIE SYMULACYJNE

Na potrzeby badania wydajności wybranych algorytmów samoorganizacji sieci sensorowych opracowano programowe narzędzie symulacyjne, pozwalające na ocenę poszczególnych algorytmów pod kątem praktycznych zastosowań w rzeczywistych sytuacjach. W oprogramowaniu zaimplementowano realizację kilku najbardziej popularnych algorytmów samoorganizacji sieci. W rozdziale 2 przedstawiono ogólnie zasady działania tych algorytmów, natomiast szczegółowe procedury znajdują się w literaturze [2-6].

Oprogramowanie do symulacji algorytmów samoorganizacji sieci sensorowych zostało zaimplementowane w języku C++ przy wykorzystaniu środowiska programistycznego C++ Builder®. Symulator umożliwia losowe lub manualne rozmieszczenie zadanej liczby węzłów, określenie ich podstawowych parametrów i wykonanie wybranych mechanizmów samoorganizacji sieci skutkujące wygenerowaniem końcowej topologii. Ponadto oprogramowanie daje możliwość zapisu/odczytu ustawień i scenariuszy symulacyjnych oraz eksportu wynikowej topologii.

Aplikacja pozwala, oprócz wyboru algorytmu samoorganizacji sieci, na ustalenie następujących parametrów symulacji: liczby węzłów sieci, maksymalnej mocy transmisji węzła, czułości odbiornika, częstotliwości pracy sieci, a także wartości kroku zmiany mocy transmisji w węzle.

4. BADANIA SYMULACYJNE

Z wykorzystaniem opisanego w punkcie 3 oprogramowania przeprowadzono szereg symulacji generowania topologii sieci sensorowych według wybranych scenariuszy, a następnie przeanalizowano właściwości energetyczne oraz struktury uzyskanych topologii. Głównymi parametrami wyznaczonymi w symulacjach, poza samą topologią, była średnia moc transmisji każdego węzła oraz sumaryczna moc transmisji wszystkich węzłów potrzebna do zachowania topologii. Dodatkowo wyznaczono średnią liczbę sąsiadów węzła w sieci, a także średnią długość ścieżek i liczbę przeskoków pomiędzy węzłami.

4.1. Założenia i parametry symulacji

Przeprowadzono serię symulacji losowo rozmieszczonej grupy węzłów o różnej liczności (25 oraz 100) na obszarze płaskiego kwadratu o boku długości 800 m. Przebadano efekty działania wybranych algorytmów i przyrównano je do sieci bez optymalizacji topologii. Podczas doboru parametrów symulacji uwzględniono parametry modułu wybranego do realizacji rozwiązania sprzętowego.

Podstawowe parametry przeprowadzonych badań symulacyjnych to: maksymalna moc transmisji węzła (3,5 dBm), zysk antenowy (0 dBi), czułość odbiornika (-94 dBm), częstotliwość pracy systemu (2400 MHz), krok zmiany mocy transmisji w węzle (0,3 dB).

Szczegółowe parametry konkretnych algorytmów:

- CBTC: $\rho=120^\circ$ oraz 60° ,
- Kneigh: $k=6\div 8$,
- LINT: $k_{\min}=4, k_{\max}=10$,
- LILT: $k_{\min}=4, k_{\max}=10$, zwiększenie mocy=50%.

W celu uproszczenia symulacji do pomiaru tłumienia propagacyjnego pomiędzy węzłami zastosowano model propagacji w wolnej przestrzeni¹ [7] (*Free Space Loss – FSL*) wyrażający się wzorem:

$$FSL_{dB} = 32,45 + 20\log(r) + 20\log(f) \quad (1)$$

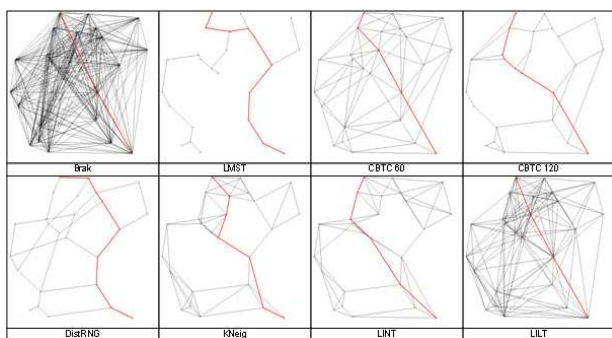
gdzie: r – odległość od nadajnika [km],
 f – częstotliwość nośnej [MHz].

Wszystkie uzyskane wyniki pozwalają na porównanie efektywności badanych algorytmów na różnych płaszczyznach, np. zużycia energii lub liczby przeskoków pomiędzy węzłami. Omówienie wybranych wyników symulacji oraz wnioski zawarto w podrozdziale 4.2.

4.2. Analiza wyników symulacji

Przeprowadzono serię badań symulacyjnych zgodnie z przedstawionymi wcześniej założeniami. Następnie uśredniono wyniki dla poszczególnych algorytmów i liczby węzłów wykorzystanych w symulacji.

Rysunek 1 ilustruje przykładowe topologie sieci wygenerowane z wykorzystaniem różnych algorytmów samoorganizacji dla 25 węzłów.



Rys. 1. Przykładowe topologie wygenerowane z wykorzystaniem różnych algorytmów samoorganizacji sieci dla 25 węzłów.

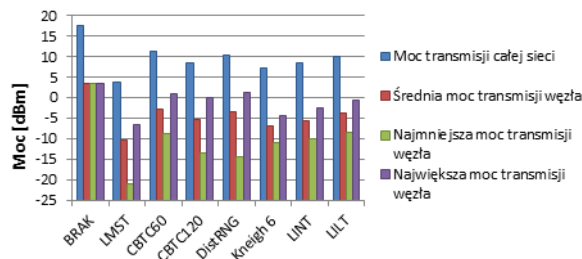
Topologie uzyskane dla poszczególnych algorytmów znacznie się różnią. Odniesieniem jest tu przypadek nie wykorzystujący samoorganizacji (każdy węzeł transmituje z mocą maksymalną). W pozostałych topologiach jest znacznie mniej aktywnych połączeń, dzięki czemu uzyskane w ten sposób sieci są dużo bardziej wydajne energetycznie. Jednak im mniej jest połączeń pomiędzy węzłami, tym mniej również alternatywnych ścieżek transmisji, w wyniku czego sieć jest mniej odporna na uszkodzenia, bądź wyczerpanie baterii węzłów.

Uzyskane wyniki liczbowe badań symulacyjnych zostały zobrazowane w postaci wykresów. W pierwszej kolejności rozpatrzono poszczególne algorytmy pod kątem mocy transmisji radiowej. Na rysunkach 2 i 3 przedstawiono zestawienie uzyskanych parametrów energetycznych. Przez moc transmisji rozumie się moc

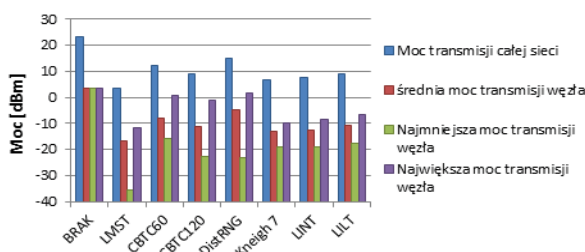
¹ Symulator nie służy analizie transmisji w sieci sensorowej, a jedynie porównaniu właściwości topologii wygenerowanych przy użyciu różnych algorytmów samoorganizacji. W tej sytuacji wybór modelu propagacyjnego nie ma istotnego wpływu na uzyskane rezultaty.

konieczną do wypromieniowania sygnału radiowego w module nadajnika węzła sieci sensorowej.

Zgodne z oczekiwaniami, w sieciach wykorzystujących algorytmy samoorganizacji można zaobserwować istotne zmniejszenie poziomów mocy transmisji sygnałów w stosunku do sieci nadającej z mocą maksymalną. Powyższa obserwacja jest prawdziwa niezależnie od gęstości sieci na danym obszarze.



Rys. 2. Zestawienie mocy transmisji dla sieci składającej się z 25 węzłów.



Rys. 3. Zestawienie mocy transmisji dla sieci składającej się z 100 węzłów.

Na podstawie analizy powyższych wykresów można zaobserwować szereg zysków płynących z wykorzystania samoorganizacji w sieci. W pierwszej kolejności należy wymienić zmniejszenie wymaganej mocy transmisji wszystkich węzłów sieci. Skutkiem tego jest nie tylko ograniczenie zużycia energii sieci jako całości, ale także ograniczenie interferencji jakie wystąpią w sieci przy dodawaniu kolejnych węzłów. Można zaobserwować również zmniejszenie wymaganych mocy transmisji (zarówno największych, jak i średnich) wraz ze wzrostem zagęszczenia węzłów na obszarze.

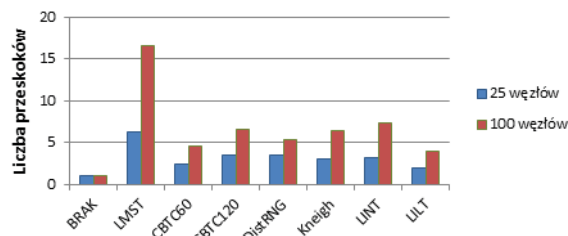
Podczas rozpatrywania parametrów energetycznych związanych z transmisją radiową, można zauważyć wyróżniający się bardzo niskimi mocami algorytm LMST. Na porównaniu przedstawionym na rysunkach 2 i 3 można zaobserwować, że posiada on znacznie lepsze parametry od wszystkich pozostałych mechanizmów. Należy mieć jednak na uwadze kształt topologii generowanej przez ten algorytm, która w wielu przypadkach prowadzi do bardzo oszczędnej w połączenia sieci i w związku z tym jest wyjątkowo wrażliwa na jakiegokolwiek uszkodzenia węzłów. Pozostałe algorytmy prezentują zbliżony względem siebie poziom wymaganych mocy transmisji. Na uwagę zasługują tutaj dobre wyniki algorytmu KNeighbours, który jest bardzo prosty w implementacji i zarazem wymaga bardzo niewiele informacji na temat sieci. Jego najistotniejszą wadą jest jednak możliwy brak połączeniowości w przypadku grupowania się węzłów w różnych lokalizacjach.

Podczas wykonanych symulacji przebadano również średnią liczbę przeskoków przy połączeniu węzłów. Wyniki przedstawiono na rysunku 4.

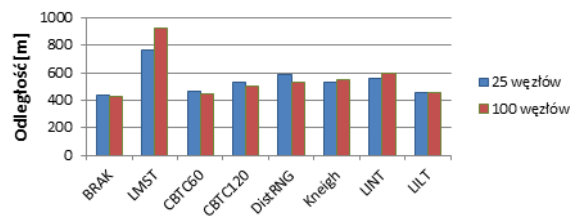
Można zauważyć, że wygenerowane topologie sieci cechujące się gęstszą siatką połączeń (przy takim samym rozmieszczeniu węzłów sieci) zazwyczaj wymagają mniejszej liczby przeskoków, aby pakiet dotarł do węzła przeznaczenia. Mniejsza liczba przeskoków jest korzystna, ponieważ prowadzi do zmniejszenia obciążenia poszczególnych węzłów podczas pracy sieci.

Wszystkie algorytmy samoorganizacji cechują się zwiększeniem liczby przeskoków względem sieci nie wykorzystujących sterowania topologią. Wśród przebadanych algorytmów wyróżnia się LMST, który cechuje się wyraźnie większą liczbą przeskoków pomiędzy węzłami. Wynika to bezpośrednio z bardzo oszczędnej w połączenia topologii, gdzie w wielu przypadkach pakiet ma tylko jedną możliwą trasę dotarcia do celu. Ta cecha algorytmu LMST jest jedną z jego najpoważniejszych wad, gdyż w sytuacji awarii któregoś z węzłów tranzycyjnych możliwość transmisji pakietu może zostać zablokowana do czasu, gdy sieć ponownie przeprowadzi samoorganizację. Dodatkowo trasy w topologii LMST są wyraźnie dłuższe niż w przypadku pozostałych algorytmów, co obrazuje rysunek 5. Może to skutkować większymi opóźnieniami w transmisji pakietów.

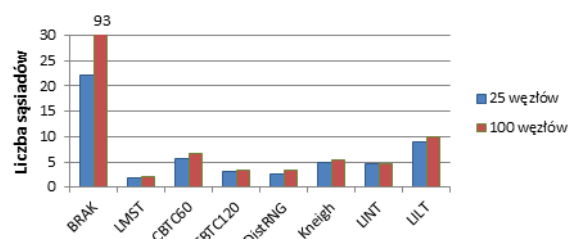
Pozostałe algorytmy cechują się zbliżonymi wartościami liczby przeskoków i długości ścieżek. Rozpatrując te rezultaty w kontekście kształtu generowanych topologii, warto zwrócić uwagę na wyniki takich algorytmów jak CBTC (z kątem 60°) i LILT. Są to algorytmy generujące topologie o względnie dużej liczbie sąsiadów węzła i sporym zagęszczeniu sieci połączeń między węzłami. Te cechy topologii pozwalają spodziewać się zwiększonej odporności sieci na utratę węzłów, jeśli tylko protokoły routingu zostaną odpowiednio dobrane.



Rys. 4. Średnia liczba przeskoków między węzłami.



Rys. 5. Średnia długość trasy pomiędzy węzłami sieci.



Rys. 6. Średnia liczba sąsiadów węzła.

Na rysunku 6 przedstawiono średnią liczbę sąsiadów węzła, która ilustruje liczbę ścieżek, którymi można przesłać informacje do kolejnego węzła na drodze do węzła docelowego.

Im większa jest średnia liczba sąsiadów, tym więcej alternatywnych ścieżek, którymi może zostać wysłany sygnał. Pod tym kątem najmniej korzystnie wypada LMST, dla którego każdy węzeł ma średnio dwóch sąsiadów (brak alternatywnej ścieżki). Średnio najwięcej sąsiadów ma węzeł w sieci z algorytmem LILT, nawet do 10. Natomiast średnia liczba sąsiadów węzła dla pozostałych algorytmów zawiera się w przedziale 3-7.

5. PODSUMOWANIE

Wyniki symulacji mechanizmów samoorganizacji bezprzewodowych sieci sensorowych przeanalizowano między innymi pod kątem przydatności tych algorytmów do realizacji w module sprzętowym zasilanym bateryjnie. W projektowanym rozwiązaniu moduły przeznaczone do pełnienia roli węzłów sieci posiadają ograniczone zasoby pamięci i możliwości przetwarzania. W związku z tym poszukiwano rozwiązania cechującego się niewielką złożonością obliczeniową, prostego w implementacji, a jednocześnie wydajnego energetycznie ze względu na bardzo ograniczone źródło zasilania.

Zdecydowano się na wybór algorytmu KNeighbors, ponieważ łączy on w sobie zalety dobrych właściwości energetycznych i prostoty implementacji. Wyróżnia się również niewielką złożonością obliczeniową, gdyż nie ma konieczności dokładnego wyznaczania pozycji węzłów ani kierunku nadejścia sygnału. Przy odpowiednim doborze liczby sąsiadów, 6 lub więcej, można ponadto zapewnić alternatywne trasy przesyłania pakietów na wypadek awarii części węzłów i w ten sposób zmarginalizować problem utraty pełnej łączności sieci.

SPIS LITERATURY

- [1] Mahalik N. P.: *Sensor Networks and Configuration. Fundamentals, Standards, Platforms and Applications*, Springer, 2007.
- [2] Li N., Hou J. C., Sha L.: *Design and Analysis of an MST-Based Topology Control Algorithm*, May 2005.
- [3] Bahramgiri M., Hajiaghayi M., Mirrokni V. S.: *Fault-tolerant and 3-Dimensional Distributed Topology Control Algorithms in Wireless Multi-hop Networks*, 2002.
- [4] Borbash S. A., Jennings E. H.: *Distributed Topology Control Algorithm for Multihop Wireless Networks*, 2002.
- [5] Blough D. M., Leoncini M., Resta G., Santi P.: *The k-Neighbors Approach to Interference Bounded and Symmetric Topology Control in Ad Hoc Network*, September 2006.
- [6] Wu K., Liao W.: *Revisiting Topology Control for Multi-Hop Wireless Ad Hoc Networks*, September 2008.
- [7] Balanis C. A.: *Antenna Theory*, John Wiley and Sons Inc., 2003.