



INSTYTUT ŁĄCZNOŚCI
PAŃSTWOWY INSTYTUT BADAWCZY

Zakład Zastosowań Technik Łączności Elektronicznej
(Z-10)

**Rozwój metod projektowania pokrycia radiowego
bezprzewodowych systemów sensorowych i lokalizacji
w oparciu o system GIS**

Praca nr 10300049

Warszawa, grudzień 2009

Rozwój metod projektowania pokrycia radiowego bezprzewodowych systemów sensorowych i lokalizacji w oparciu o system GIS

Wniosek nr 1006/09

Praca nr 10.300.04.9

Słowa kluczowe: sensory, sieci sensorowe, Geographical Information System (GIS)

Kierownik pracy: dr inż. Jacek Jarkowski

Wykonawcy: dr inż. Jacek Jarkowski
mgr inż. Konrad Łakomiec
Katarzyna Godlewska

Zakład Zastosowań Technik Łączności Elektronicznej

Kierownik zakładu: inż. Bogdan Chojnacki

WSTĘP.....	4
1. BUDOWA SENSORA – WĘZŁA SIECI SENSOROWEJ.....	5
2. ARCHITEKTURA SIECI SENSOROWEJ.....	6
3. PROTOKOŁY KOMUNIKACYJNE – MODEL WARSTWOWY SIECI SENSOROWYCH.....	7
4. RUTING W SIECIACH SENSOROWYCH.....	9
4.1. PŁASKIE PROTOKOŁY RUTINGU.....	11
4.1.1. <i>Sensor Protocol for Information via Negotiation – SPIN</i>	13
4.1.2. <i>Directed Diffusion</i>	14
4.1.3. <i>Rumor Ruting</i>	16
4.1.4. <i>Random Walks</i>	17
4.1.5. <i>Energy Aware Ruting</i>	17
4.1.6. <i>Minimum Cost Forwarding Algorithm</i>	18
4.2. HIERARCHICZNE PROTOKOŁY RUTINGU.....	18
4.2.1. <i>Low-Energy Adaptive Clustering Hierarchy</i>	18
4.2.2. <i>Power-Efficient Gathering in Sensor Information Systems</i>	22
4.2.3. <i>Threshold sensitive Energy Efficient sensor Network</i>	22
4.2.4. <i>Adaptive Threshold sensitive Energy Efficient sensor Network</i>	23
4.3. LOKALIZACYJNE PROTOKOŁY RUTINGU.....	23
5. WYKORZYSTANIE TECHNOLOGII GIS W PROJEKTOWANIU SIECI SENSOROWYCH	24
6. WYZNACZANIE POŁOŻENIA WĘZŁÓW SIECI SENSOROWEJ	27
7. ZASTOSOWANIE SYSTEMU GIS W LOKALIZACJI WĘZŁÓW SIECI SENSOROWEJ	30
SPIS RYSUNKÓW	34
BIBLIOGRAFIA.....	35

Wstęp

Technika sensorowa budzi w ostatnim czasie coraz większe zainteresowanie z uwagi na możliwość monitorowania zmian zachodzących w środowisku. Sensory tworzące bezprzewodową sieć w zależności od swojej lokalizacji i przeznaczenia mogą kontrolować takie wielkości jak: temperatura, wilgotność, ruch, ciśnienie, stopień zanieczyszczenia powietrza itp. Ograniczone zasoby energetyczne i możliwości obliczeniowe sensorów wymagają odpowiedniej organizacji sieci pod względem zapewnienia:

- **łączności** – możliwości komunikacji pomiędzy sensorami,
- **pokrycia** – możliwości monitorowania wyznaczonego obszaru.

Projektowanie bezprzewodowych sieci sensorowych wiąże się z takimi problemami jak: rozmieszczenie sensorów w taki sposób, aby zapewnić odpowiednie pokrycie monitorowanego obszaru, pozyskanie informacji o położeniu sensorów i ich późniejszej pracy. W rozwiązywaniu powyższych problemów projektowych niezmiernie przydatny będzie system GIS.

1. Budowa sensora – węzła sieci sensorowej

Sensor jest kluczowym elementem sieci. W jego architekturze możemy wyróżnić cztery moduły(rys.1):

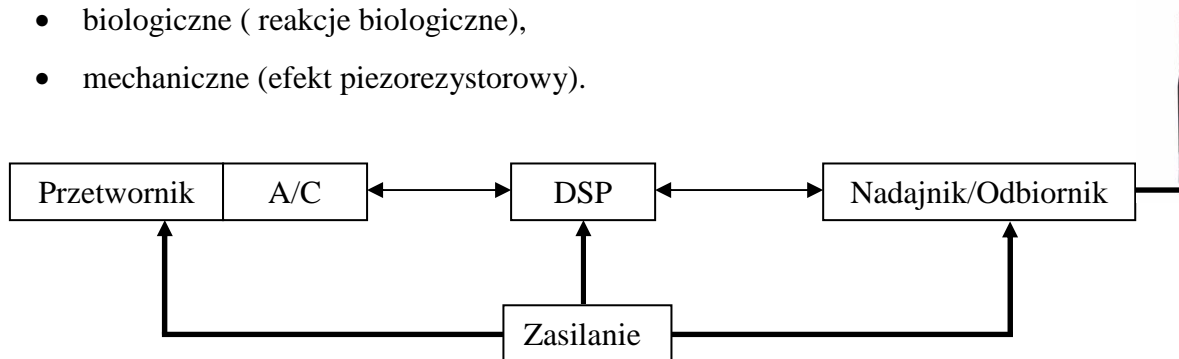
- pomiarowy,
- obliczeniowy,
- transmisyjny,
- zasilania.

Głównym elementem modułu pomiarowego jest przetwornik, który generuje określony sygnał analogowy pod wpływem zmian zachodzących w monitorowanym przez niego otoczeniu. Uzyskany sygnał z przetwornika obrabiany jest w mikrokontrolerze bądź w mikroprocesorze (moduł obliczeniowy) i przesyłany drogą radiową do innych węzłów(moduł transmisyjny).

Najsłabszym ogniwem węzłów sensorowych jest moduł zasilający. Sensory czerpią zasilanie głównie z baterii, dlatego w niektórych zastosowaniach sieci, (gdzie wymiana baterii jest niemożliwa, np. w przypadku rozrzucenia sensorów z samolotu w niedostępne miejsca) bardzo istotny jest problem oszczędzania energii. Ponadto z uwagi na potrzebę miniaturyzacji tego typu urządzenia mają ograniczone moce obliczeniowe. Pomimo tych ograniczeń siła sieci tkwi w ilości sensorów oraz informacji, jakie mogą pozyskać, wstępnie przetworzyć i przesłać dalej.

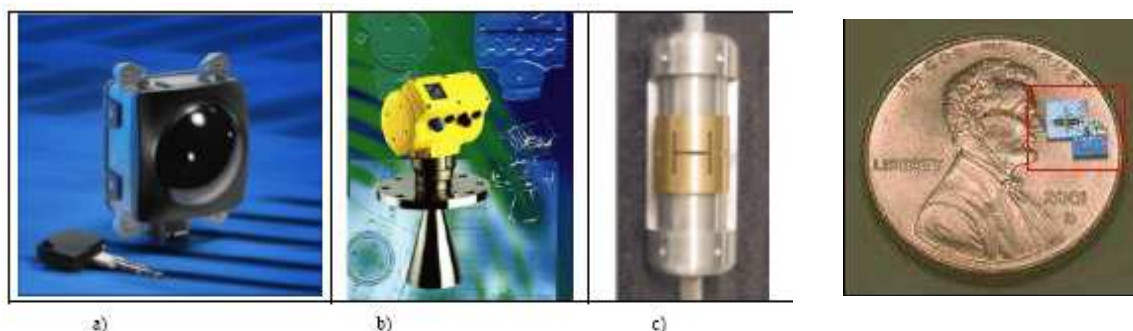
W zależności od efektu, zjawiska fizycznego, na którym oparte jest działanie przetworników, możemy wyróżnić sensory:

- termiczne (efekt Seebecka,),
- magnetyczne, elektromagnetyczne (efekt Halla),
- optyczne (efekt fotoelektryczny),
- chemiczne (wykorzystanie reakcji chemicznych),
- biologiczne (reakcje biologiczne),
- mechaniczne (efekt piezorezystorowy).



Rys. 1. Architektura węzła sensorowego.

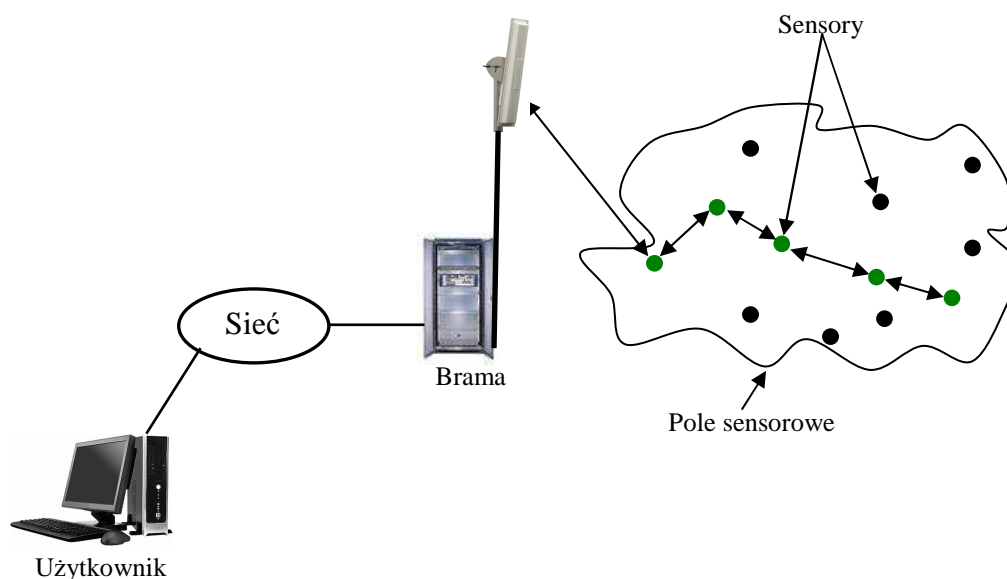
Przykładowe sensory(czujniki) przedstawiono na rys.2 Trzeba mieć tutaj świadomość tego iż dzięki miniaturyzacji urządzeń elektromechanicznych możliwe jest stosowanie w sieciach węzłów o niewielkich rozmiarach (rys.2).



Rys. 2. Przykłady sensorów: a) radar samochodowy, b) czujnik pomiaru cieczy w zbiornikach, c) miernik temperatury linii przesyłowej 420 kV –TU Darmstadt

2. Architektura sieci sensorowej

Uproszczoną architekturę sieci sensorowej przedstawia rys.3



Rys. 3. Uproszczona architektura sieci sensorowej.

Na danym obszarze (tzw. polu sensorowym) sensory mogą być rozmieszczone manualnie przez człowieka bądź rozrzucone z samolotu lub w inny sposób. Sensory mają za zadanie zbierać informacje (monitorować dany obiekt, wykonywać pomiary), a następnie przesyłać je do tzw. bramy (węzła nadrzędnego). W komunikacji pomiędzy określonym sensorem położonym stosunkowo daleko od bramy muszą pośredniczyć inne sensory (mały zasięg, ograniczona moc), więc przesyłanie danych jest realizowane skokowo. Dane z węzła nadrzędnego kierowane są dalej przez sieć do użytkownika

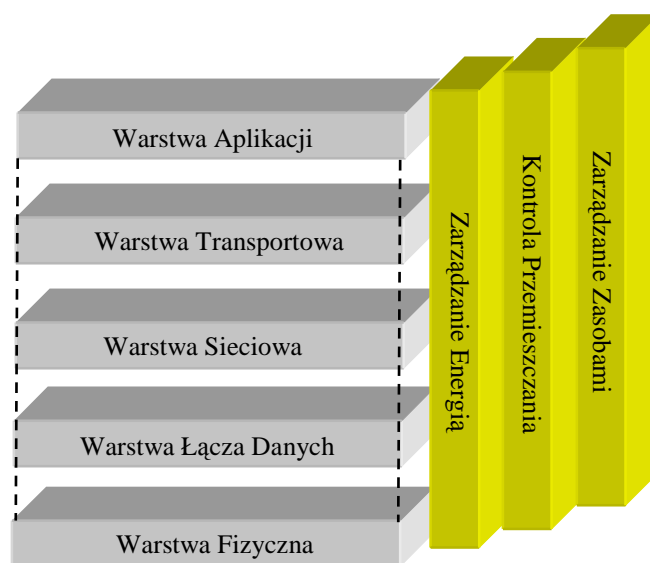
Sieć sensorowa skupia dużą liczbę węzłów, które często mogą pracować w niedostępnych miejscach uniemożliwiających serwis (np. wymianę baterii). Pozostawienie sensora bez kontroli sprzyja powstawaniu różnorodnych usterek, które wymuszają częste zmiany topologii sieci. Częsta rekonfiguracja sieci wymaga odpowiednio dużej liczby węzłów oraz specjalnych protokołów routingu.

3. Protokoły komunikacyjne – model warstwowy sieci sensorowych

Algorytmy i protokoły stosowane w tradycyjnych sieciach ad-hoc czy też sieciach komórkowych nie są odpowiednie dla sieci sensorowych. Sieci sensorowe różnią się od sieci bezprzewodowych ad-hoc tym, iż:

- mogą mieć o kilka rzędów wielkości większą liczbę węzłów niż ad-hoc (*uniemożliwia to wprowadzenie globalnych adresów IP z uwagi na dużą ilość informacji*),
- węzły sieci są gęsto rozmieszczone i nie i często wykazują brak odporności na awarie,
- węzły sieci dysponują małą mocą obliczeniową, pamięcią oraz energią (*konieczne jest sprawne zarządzanie zasobami*),
- sieć zmienia topologie bardzo często (*konfiguracja i organizacja musi być realizowana automatycznie*),
- stosowana jest komunikacja broadcast, natomiast w sieciach ad-hoc point to point.

Analizując protokoły routingu używane w sieci sensorowej wprowadza się jej model warstwowy zilustrowany na rys.4. Przedstawiona struktura składa się z kilku warstw m.in.: fizycznej, łącza danych, sieciowej, transportowej, aplikacji (rys.4)



Rys. 4. Struktura (model) warstwowa sieci sensorowej [1]

Warstwa fizyczna jest odpowiedzialna za komunikację bezprzewodową pomiędzy sensorami, czyli za wybór częstotliwości, modulację, detekcję oraz szyfrowanie. W komunikacji między węzłami występują niekorzystne zjawiska propagacyjne, podobnie jak w innych systemach bezprzewodowych, czyli odbicie fali nośnej, dyfrakcja, zanik sygnału oraz wielodrogowość. Oczywiście wpływ tych zjawisk możemy ograniczyć zarówno poprzez odpowiednie rozwiązania sprzętowe, jak również stosując większą liczbę sensorów w sieci. Większe zagęszczenie sensorów pozwala nie tylko zmniejszyć wpływ niekorzystnych zjawisk związanych z propagacją wielodrogową, ale także poprawić energooszczędność sensorów.

Warstwa łączy danych realizuje zadania związane z multipleksowaniem, detekcją ramek danych, kontrolą błędów oraz przydzielaniem dostępu do nośnika. Dostęp do medium realizowany jest na bazie protokołów opierających się na technice: TDMA (ang. *Time Division Multiple Access*), TDMA/FDMA (ang. *Time Multiple Access/Frequency Division Multiple Access*) i CDMA (ang. *Code Division Multiple Access*).

Warstwa sieciowa odpowiada za proces routingu danych z warstwy transportowej, natomiast sama warstwa transportowa łączy warstwę aplikacji i sieci poprzez realizowanie procesów multipleksacji i demultipleksacji oraz dostarczania rozwiązań (mechanizmów) przesyłania danych pomiędzy bramą (węzłem nadrzędnym) a sensorami. Ruch głównie prowadzony jest od sensorów do bramy (przesyłanie pomiarów), natomiast ścieżką w odwrotnym kierunku przesyłane są informacje zarządzające siecią (komendy pomiarów, zapytań). W warstwie transportowej przesyłanie informacji pomiędzy bramą a sensorami prowadzone jest w oparciu o protokoły[2]:

- **ESRT** (ang. *Event-to-Sink Reliable Transport*) – zaimplementowany jest w bramie (węźle nadrzędnym) i wykorzystywany do transportu danych pomiarowych z sensorów do węzła nadrzędnego. Dzięki takiemu rozwiązaniu zasoby sensorów są mniej obciążone. Powyższy algorytm zmieniając częstotliwość zapytań sensorów bada możliwość zaistnienia danego zdarzenia.
- **PSFQ** (ang. *Pump Slowly Fetch Quickly*) – używany jest do rozsyłania (transportu) zapytań, komend z bramy do sensorów. Pozwala odzyskać utracone dane, dlatego też sensory muszą dysponować odpowiednią pamięcią, aby w kolejnej fazie transmisji przesłać utracone informacje.

W warstwie aplikacji wykorzystywane są trzy podstawowe protokoły:

- **SMP** (ang. *Sensor Management Protocol*) – umożliwia komunikację z siecią sensorową za pośrednictwem np. sieci Internet, a co za tym idzie pozwala włączyć i wyłączyć

dany sensor oraz sprawdzić jego konfigurację pomiarową. Ponadto zapewnia synchronizację między sensorami.

- **TADAP** (ang. *Task Assignment and Data Advertisement Protocol*) – wykorzystywany jest do wydawania poleceń sensorom w postaci zapytań (adresowanych do grupy bądź jednego węzła), np. *podaj bądź wykonaj pomiar poziomu wody.*
- **SQDDP** (ang. *Sensor Query and Data Dissemination Protocol*) – podobnie jak TADAP umożliwia wydawanie poleceń w postaci zapytań, które dodatkowo uzupełnione zostały w określone atrybuty danych (czy też lokalizacyjne), np. *podanie lokalizacji gdzie w miejscu rozmieszczenia sensorów (załóżmy wzdłuż odcinka monitorowanej rzeki) poziom wody przekroczył stan alarmowy o 1m.*

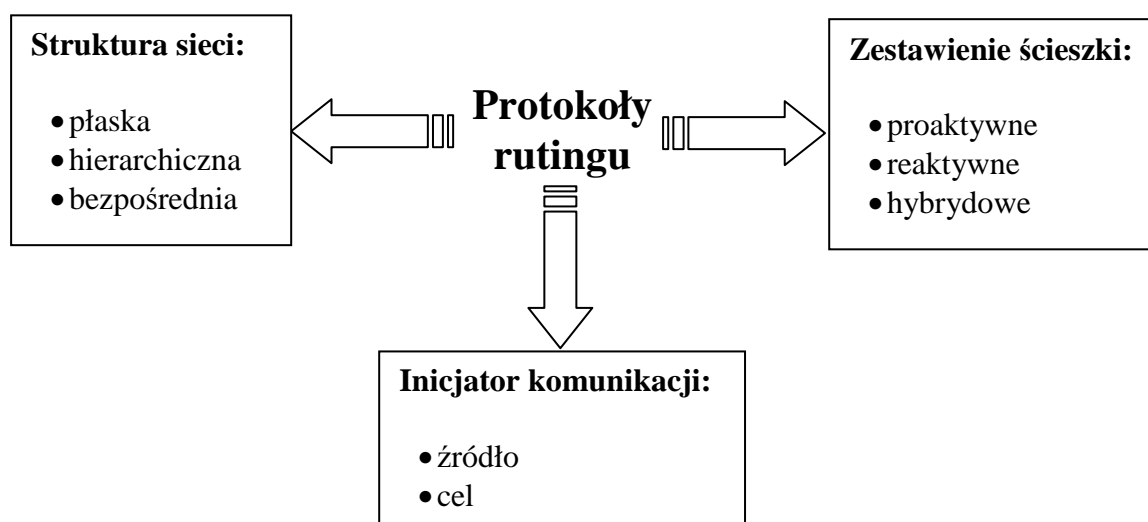
W modelu sieci sensorowej zobrazowanej na rys.4 oprócz opisanych wyżej warstw możemy również wyróżnić takie płaszczyzny jak:

- *Zarządzania energią* – kontroluje wykorzystanie energii przez sensor (np. gdy bateria sensora jest na wyczerpaniu, wówczas może on nie brać udziału w przesyłaniu informacji od innych węzłów, zaś wykorzystać pozostałą energię na wykonywanie własnych zadań pomiarowych).
- *Kontroli przemieszczania* – wykorzystywana jest w zastosowaniach mobilnych sieci sensorowych (pozwala rejestrować pozycję sensorów – dzięki temu węzły w danej chwili mają informację o swoich nowych sąsiadach, ich możliwościach energetycznych).
- *Zarządzania zasobami* – planuje i nadzoruje pomiary wybranych sensorów (nie wszystkie węzły na danym obszarze, w tej samej chwili muszą realizować pomiary) biorąc pod uwagę indywidualne zasoby energetyczne węzłów.

4. Ruting w sieciach sensorowych

Protokoły routingu wykorzystywane w sieciach sensorowych można podzielić ze względu na (rys.5):

- Struktūrę sieci,
- Zestawienie ścieżki,
- Inicjatora komunikacji.



Rys. 5. Podział protokołów routingu w sieciach sensorowych

Biorąc pod uwagę strukturę sieci możemy wyróżnić protokoły bezpośrednie, płaskie oraz hierarchiczne. W przypadku struktury bezpośredniej, w komunikacji pomiędzy sensorem a węzłem nadrzędnym **nie pośredniczą inne węzły** (sensory). Komunikacja bezpośrednia może być niekiedy nie możliwa ze względu na duże oddalenie sensora od węzła nadrzędnego, dlatego też jest ona dosyć niepraktyczna nie tylko z uwagi na odległość, ale też ze względów energetycznych. W strukturze płaskiej komunikacja najbardziej oddalonego sensora z węzłem nadrzędnym jest realizowana **za pośrednictwem innych węzłów** (mniejsza moc nadawania, mniejsze zużycie energii) – węzeł najbliższy łączy się z tzw. bramą. Protokoły struktury hierarchicznej tworzą z rozmieszczonych węzłów na określonym obszarze tzw. klastry, czyli grupy sensorów wśród, których jeden z nich pełni funkcję węzła głównego komunikującego się z bramą. Sensory wchodzące w skład określonego klastra komunikują się tylko i wyłącznie z węzłem głównym, zaś on łączy się bezpośrednio z bramą (węzłem nadrzędnym).

W podziale uwzględniającym inicjatora komunikacji można wyróżnić protokoły bazujące na źródle oraz celu (przeznaczeniu) informacji (zapytania oraz dane pomiarowe). Źródłem danych jest sensor, natomiast celem węzeł nadrzędny. Protokoły inicjowane przez sensor przesyłają informacje do bramy (kiedy zaszło określone zdarzenie np. przekroczony został stan alarmowy mierzonego parametru w danym punkcie pomiarowym), które raportowane są w sposób ciągły lub zdarzeniowy.

Biorąc pod uwagę sposób zestawiania ścieżki routingu można wyodrębnić trzy protokoły: proaktywne, reaktywne oraz hybrydowe. Protokoły proaktywne przechowują wszystkie dostępne ścieżki w tablicach routingu, które muszą posiadać wszystkie sensory sieci.

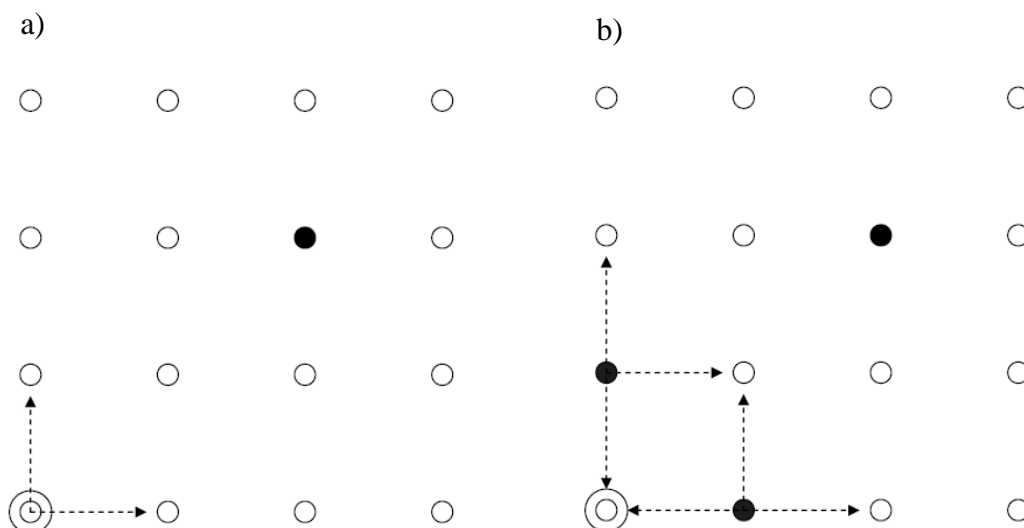
W przypadku zmiany topologii sieci bądź wyłączenia określonych sensorów z pośrednictwa w komunikacji z bramą (z powodu niskiego poziomu energii niezbędnego do prowadzenia tylko własnych pomiarów) zachodzi potrzeba wyznaczania nowych ścieżek, które w postaci nowej tabeli routingu muszą być rozpropagowane do wszystkich węzłów sensorowych, co w sytuacji dużej sieci z ogromną liczbą węzłów jest uciążliwe. Ponadto dochodzi tutaj problem zarządzania samą tablicą routingu, dlatego też tego typu protokoły nie są zbyt dobre dla sieci składającej się z dużej liczby sensorów. Nie co inaczej funkcjonują protokoły reaktywne, które wyliczają ścieżkę komunikacji między sensorem a węzłem nadrzędnym w sposób dynamiczny (gdy konieczna będzie transmisja). Natomiast protokoły hybrydowe bazują na rozwiązaniach omówionych pokrótce protokołów proaktywnych i reaktywnych. Reasumując najpierw wyznaczają wszystkie ścieżki (tworzą tabele routingu), a dopiero później w miarę konieczności dokonują ich modyfikacji (udoskonalenia).

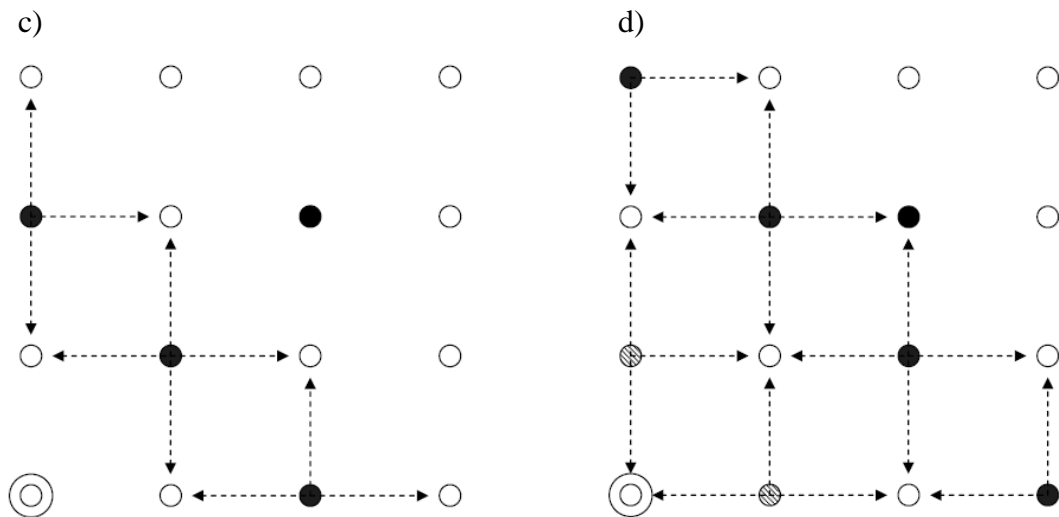
Na ogół na pierwszym miejscu funkcjonuje podział na protokoły płaskie i hierarchiczne a dopiero można mówić o podziale uwzględniającym inicjatora komunikacji. Jeśli chodzi o podział uwzględniający sposób zestawiania ścieżki to w praktyce przeważają protokoły reaktywne. W dalszej części pracy zostaną omówione protokoły płaskie i hierarchiczne.

4.1. Płaskie protokoły routingu

Początkowo dla potrzeb sieci sensorowych zaadaptowano dosyć proste algorytmy (**flooding**, **gossiping**) [3] rozsyłania danych stosowane w sieciach ad-hoc.

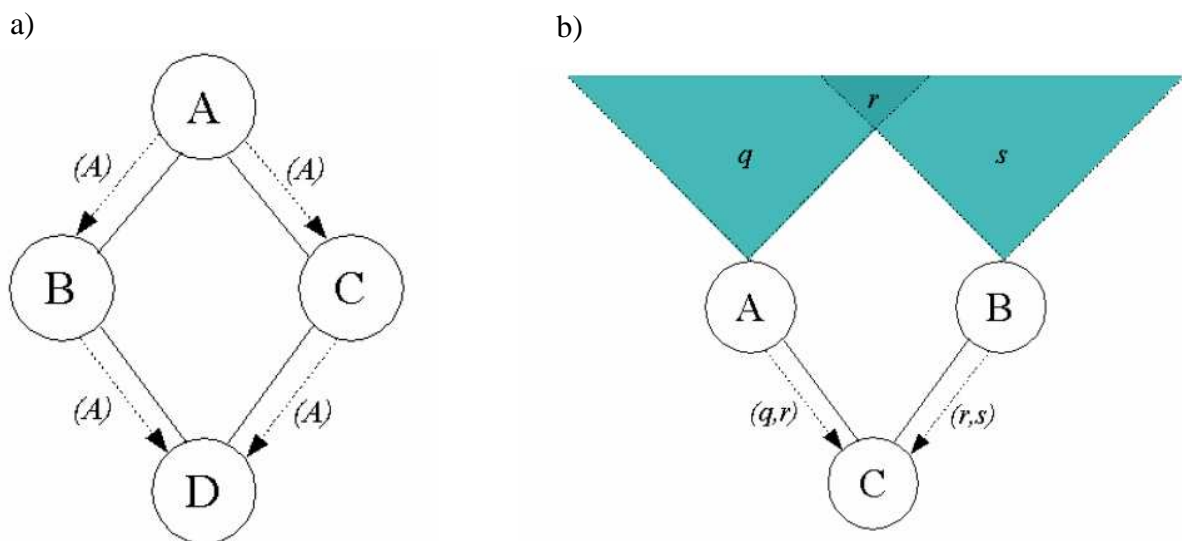
W protokole **flooding** każdy sensor do którego zostały wysłane dane (np. pomiarowe) zapisuje je, a następnie wysyła do wszystkich swoich sąsiadów. Reasumując opiera się na tzw. „zalewaniu” (rys. 6)





Rys. 6. Ilustracja działania protokołu flooding (zalewania)

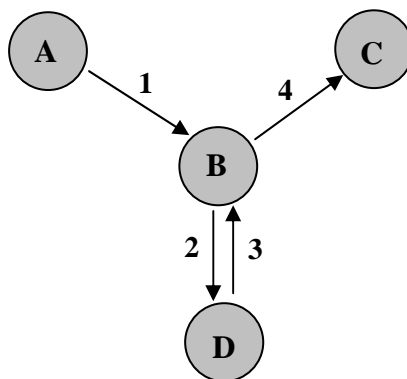
Protokół flooding pozwala stosunkowo szybko rozsyłać dane, ale niestety jest on dobry tylko dla sieci, które dysponują dużymi zasobami energetycznymi oraz stosunkowo dużym pasmem. Ponadto z uwagi na szereg wad min. implozję (powtarzanie się informacji otrzymywanych przez sensor – rys. 7a), overlap (pokrywanie się obszarów monitorowanych przez sensory – rys. 7b). nie nadaje się do używania w sieciach sensorowych.



Rys. 7. Ilustracja wad protokołu flooding: a) Implozja, b) Overlap [4]

Protokół gossiping stanowi ulepszoną wersję flooding. W tym protokole sensor (węzeł) nie propaguje danych do wszystkich swoich sąsiadów, lecz do jednego bądź kilku wybranych w drodze losowania. W trakcie procesu losowania istnieje możliwość wybrania także sensora, który przysłał dane. Ponadto sensor po otrzymaniu danych odsyła je do węzła, z którego napłynęły (rys. 8). Jest to konieczne ponieważ niektóre węzły nie otrzymały by pożądaných

danych. Należy tutaj podkreślić, że zarówno w gossiping jaki i fooding węzły tworzą kopie danych i dopiero wysyłają je dalej. W przypadku gossiping liczba kopi jest mniejsza co redukuje nieco zjawisko implozji, ale za to jego wadą jest długi czas rozsyłania informacji (danych).



Rys. 8. Ilustracja działania protokołu gossiping

4.1.1. Sensor Protocol for Information via Negotiation – SPIN

Protokół SPIN został opracowany na podstawie **fooding**, który jak wspomniano wyżej cechuje się kilkoma poważnymi wadami (duplikacja informacji, węzłów oraz brak sprawdzania zasobów sensora). Oczywiście w SPIN wprowadzono mechanizmy likwidujące wymienione wady min: **negocjacje przed rozpoczęciem transmisji oraz zarządzanie zasobami**. Pierwszy mechanizm umożliwia eliminację nadmiarowych informacji poprzez rozsyłanie treści opisujących ich zawartość (dzięki temu sensor odbiorczy może je odebrać lub też nie). Z kolei drugi pozwala kontrolować (analizować) zasoby energetyczne węzłów, a co zatem idzie sterować ich aktywnością w przypadku małej zasobności w energię.

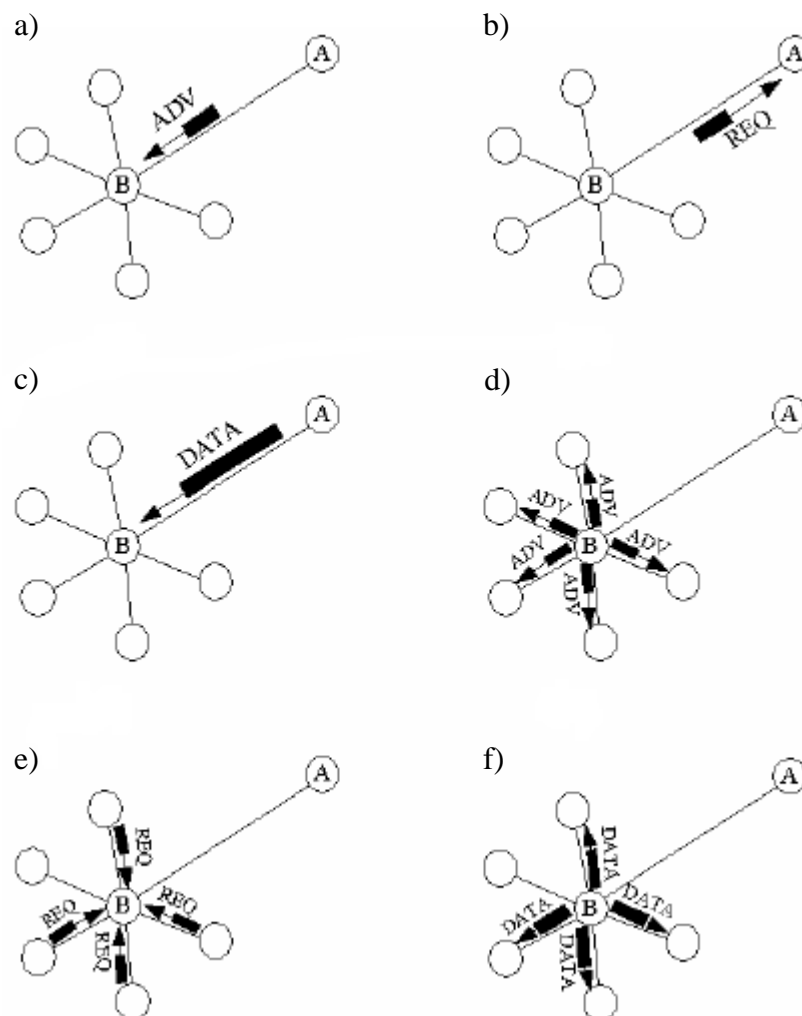
W komunikacji protokół SPIN używa trzy typy wiadomości:

- **ADV** – węzeł rozgłasza informację o danych (opis informacji do przesłania),
- **REQ** – wyraża (zgłasza) chęć transmisji danych,
- **DATA** – dane.

Protokół rozpoczyna swoje działanie gdy do sensora napłyną nowe dane i będzie je chciał przesłać dalej. W takiej sytuacji sensor wysyła do swoich sąsiadów komunikat ADV, a gdy dany sąsiad wyrazi chęć transmisji wówczas wysyła wiadomość REQ. Po komunikacji REQ rozpoczyna się transmisja danych DATA. Istotę działania protokołu SPIN zobrazowano na rys. 9.

Naturalnie nie jest to protokół bez wad otóż gdy węzły pośredniczące w transmisji pomiędzy źródłem wiadomości (określonym sensorem) a węzłem nadrzędnym nie wyrażą

chęci otrzymania danych to wówczas może wystąpić sytuacja, że one nigdy nie zostaną dostarczone do bramy. Trzeba tutaj podkreślić, że SPIN raczej sprawdza się w małych sieciach sensorowych.



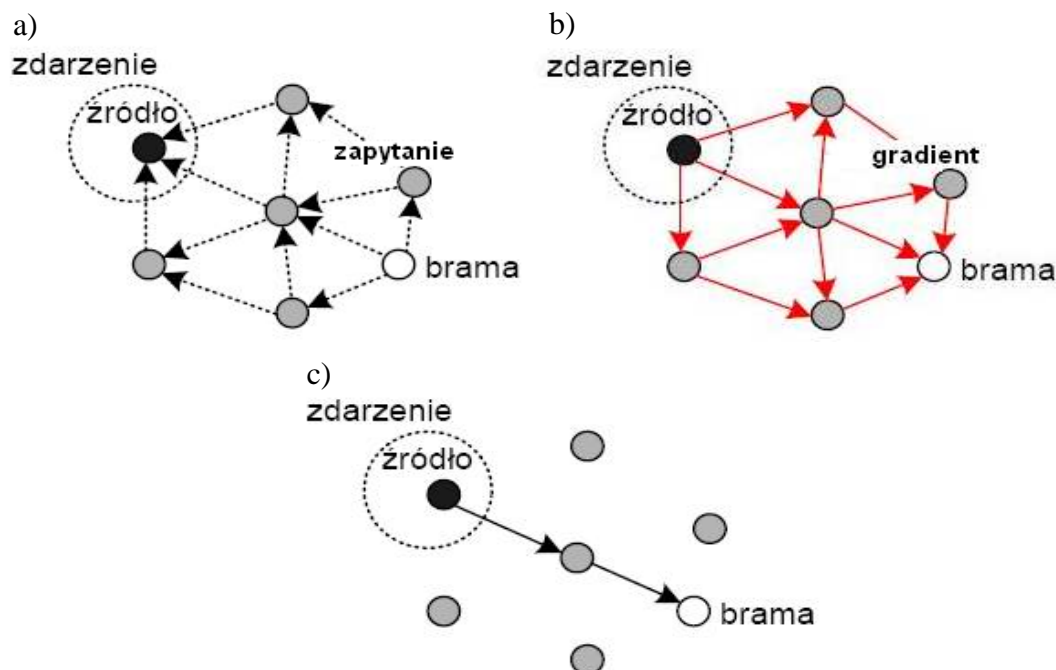
Rys. 9. Działanie protokołu SPIN[4]

4.1.2. Directed Diffusion [5]

W protokole Directed – Diffusion użytkownik chcąc uzyskać określone dane z sieci sensorowej musi utworzyć odpowiednie *zapytanie*, które będzie opisywało poszukiwane przez niego informacje. Szukane dane w zapytaniu najczęściej opisywane są w postaci: **atrybut – wartość**. Sformułowane w ten sposób zapytanie kierowane jest poprzez Internet bądź też inne łącze (np. satelitarne) do bramy, a następnie w określony region sieci.

W trakcie dystrybucji *zapytania* po sieci, w każdym sensorze konstruowany jest *gradient* wskazujący kierunek sensora z którego nadeszło zapytanie oraz tworzone jest wiele ścieżek od bramy do węzła docelowego. Z pośród wielu ścieżek do transmisji wybierana jest tylko jedna aby nadmiernie nie obciążać sieci. Idę działania protokołu zilustrowano na rys.10.

W zależności od wykonywanych zadań (kontrola określonych zjawisk na danym terenie, stanów) zapytanie do poszczególnych sensorów może być rozprowadzane na podstawie np. współrzędnych, czy też innych atrybutów zależnych od aplikacji użytkownika sieci.



Rys. 10. Działanie protokołu Directed Diffusion: a) dystrybucja zapytania, b) tworzenie gradientów, c) transmisja danych [5]

Jeśli przyjmiemy, że dana sieć sensorowa monitoruje np. stan wody wzdłuż koryta rzeki i chcemy mieć informacje czy kontrolowana rzeka na danym odcinku przekroczyła stan alarmowy, to wówczas przykładowe zapytanie może mieć postaci:

Stan = 1m – określa próg wykonywanych pomiarów (podaj dane jeżeli stan alarmowy został przekroczony o 1m),

Przedział = 3 min – po przekroczeniu stanu alarmowego wykonuj pomiary i je wysyłaj co 3min,

Obszar = [51.10, 15.5, 51.20, 16.5] – informacje mają być zbierane z danego regionu (np. współrzędne rozmieszczenia sensorów naniesione na mapę GIS).

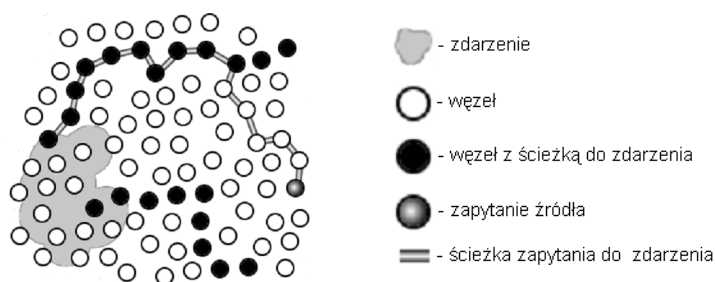
Sensor po otrzymaniu powyższego zapytania sprawdza czy posiada w swojej podręcznej pamięci wymagane informacje, a jeśli nie to czy będzie w stanie wykonać pomiar. Gdy nie jest możliwe wykonanie pomiarów to wówczas przesyła zapytanie dalej. Niezależnie od tego czy sensor posiada informacje spełniające warunki w *zapytaniu* ustawia gradient w kierunku sensora od którego otrzymał *zapytanie*. Gdy sensor pośredniczący w transmisji

odbierze dane skierowane do bramy to poszukuje w swojej pamięci wpisy, które będą odpowiadały zapytaniu oraz nadesłanym danym. Gdy ich nie znajdzie kasuje otrzymane dane, w innym przypadku przesyła je dalej.

4.1.3. Rumor Ruting [6]

Protokół Rumor Ruting bazuje na rozwiązaniach przyjętych w protokole Directed Diffusion, z tym iż przekazuje on *zdarzenia* a nie *zapytania*. W powyższym protokole zapytania kierowane są tylko i wyłącznie do sensorów, które monitorują dane zdarzenie opisane w zapytaniu. Sensor który zaobserwuje (zmierzy) dane zdarzenie opisane w zapytaniu zapisuje (dodaje) je w swojej tablicy zdarzeń oraz generuje specjalny pakiet określany mianem *agenta*. W pakiecie agenta zawarte jest specjalne pole TTL (ang. *time to live*), które określa czas jego życia w sieci oraz tabele z pomiarami (zdarzeniami) zmierzonymi (zaobserwowanymi) przez sensor. Utworzony agent jest przesyłany przez sieć do węzła nadrzędnego (bramy). Wszystkie sensory, które odebrały pakiet agenta aktualizują swoją tablice zdarzeń. Agent transmitowany jest przez sieć w ten sposób, że jeśli odbierze go nowy sensor to wówczas wyszukuje on sąsiada tego węzła (sensora), który nie widnieje na liście dotychczasowo odwiedzonych sensorów (każdy nowo wybrany sensor zmniejsza wartość pola TTL o 1). Natomiast zapytania kierowane przez bramę w celu uzyskania informacji - czy dane zdarzenie zaistniało? są w sposób losowy przesyłane do kolejnych sensorów aż odnaleziona zostanie ścieżka do pożądanego zdarzenia (rys. 11).

Powyższy protokół może okazać się mało oszczędny energetycznie w sytuacji gdy liczba zdarzeń będzie zdecydowanie większa od zapytań, wówczas pakiet agenta będzie bardzo duży ze względu na powiększającą się tablice zdarzeń. Ponadto może wystąpić kolejny problem wynikający z tego iż pakiet agenta wysyłany do bramy zawiera informacje tylko o jednej ścieżce do określonego zdarzenia. W takiej sytuacji sensory tworzące ścieżkę do danego zdarzenia mogą być w przypadku dużej liczby zapytań nadmiernie obciążone energetycznie (swoje zasoby energetyczne mogą wyczerpać stosunkowo szybko).



Rys. 11. Działanie protokołu Rumor Ruting

4.1.4. Random Walks [7]

Protokół *Random Walks* realizuje ruting reaktywny i przeznaczony jest głównie dla sieci dużych o nie przemieszczających się węzłach. Każdy z sensorów ma nadany unikalny identyfikator (ID) po którym jest rozpoznawany. W *Random Walks* optymalna ścieżka pomiędzy wybranym sensorem a bramą wyszukiwana jest w oparciu o uogólniony algorytm Dijkstry. Jeżeli topologie sieci sensorowej będziemy rozpatrywać na płaszczyźnie grafu (miejsca przecięcia się siatki na płaszczyźnie wyznaczają położenie sensorów) to algorytm Dijkstry umożliwi znalezienie nam najkrótszej ścieżki biorąc pod uwagę tylko i wyłącznie dodatnie wagi krawędzi grafu odpowiadające odległości między sensorami. W przypadku uogólnienia tego algorytmu pod uwagę brane są także wagi ujemne. Reasumując dany sensor spośród wszystkich swoich sąsiadów wybiera z pewnym prawdopodobieństwem tego, który jest najbliżej określonego celu (węzła docelowego lub bramy).

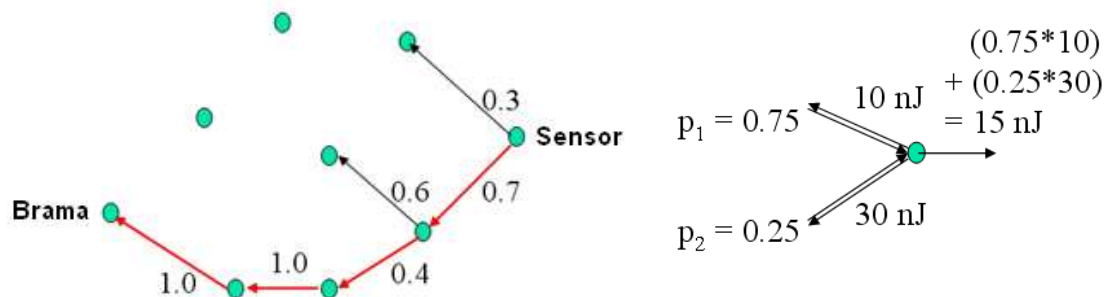
4.1.5. Energy Aware Ruting [8]

W protokole Energy Aware Routing w przeciwieństwie do Directed Diffusion dane pomiędzy określonym sensorem a bramą przesyłane są różnymi możliwymi ścieżkami. Takie rozwiązanie umożliwia bardziej równomierne zużycie energii sensorów.

Działanie protokołu Energy Aware Routing można podzielić na trzy fazy:

- tworzenia ścieżki,
- transmisji danych,
- administracja ścieżkami.

W trakcie tworzenia (zestawiania) ścieżki konstruowane są dla każdego sensora tablice rutingu (wykorzystywany jest protokół flooding). Należy tutaj podkreślić, że zestawiane ścieżki są wybierane do transmisji losowo w zależności od przypisanego im prawdopodobieństwa odzwierciedlającego zasoby energetyczne poszczególnych węzłów. Sposób zestawiania ścieżki obrazuje rys. 12.



Rys. 12. Zestawianie ścieżki w protokole Energy Aware Ruting

W kolejnych etapach realizowana jest transmisja danych oraz zarządzanie ścieżkami.

4.1.6. Minimum Cost Forwarding Algorithm [9]

W protokole Minimum Cost Forwarding Algorithm (MCFA) węzły nie tworzą tablicy routingu, lecz wyliczają wartość minimalnego kosztu ścieżki od siebie do bramy. Brama na początku ustawia wartość tego kosztu na zero, natomiast sensory ustawiają swoje koszty na nieskończoność. W kolejnym kroku brama rozsyła do węzłów informację o koszcie swojej ścieżki. Sensor po odebraniu tej informacji dokonuje aktualizacji kosztu ścieżki (jeśli jest on mniejszy od poprzedniego to rozsyła tą wiadomość do innych węzłów). Koszt danej ścieżki może być liczony na podstawie opóźnienia bądź też liczby sensorów.

Gdy sensor zamierza wysłać swoje dane pomiarowe do bramy to rozsyła je do swoich sąsiadów, następnie każdy z sąsiednich sensorów po otrzymaniu danych sprawdza czy znajduje się na ścieżce do bramy, której koszt jest najmniejszy. Jeśli okaże się, że tak to wówczas rozsyła on dane z kolei do swoich sąsiadów. Powyższa operacja powtarza się do momentu aż dane otrzyma brama.

W tym protokole może niekiedy dochodzić do nadmiernego zużycia energii niektórych sensorów. Ten problem wynika z braku systematycznej (okresowej) aktualizacji kosztów ścieżek.

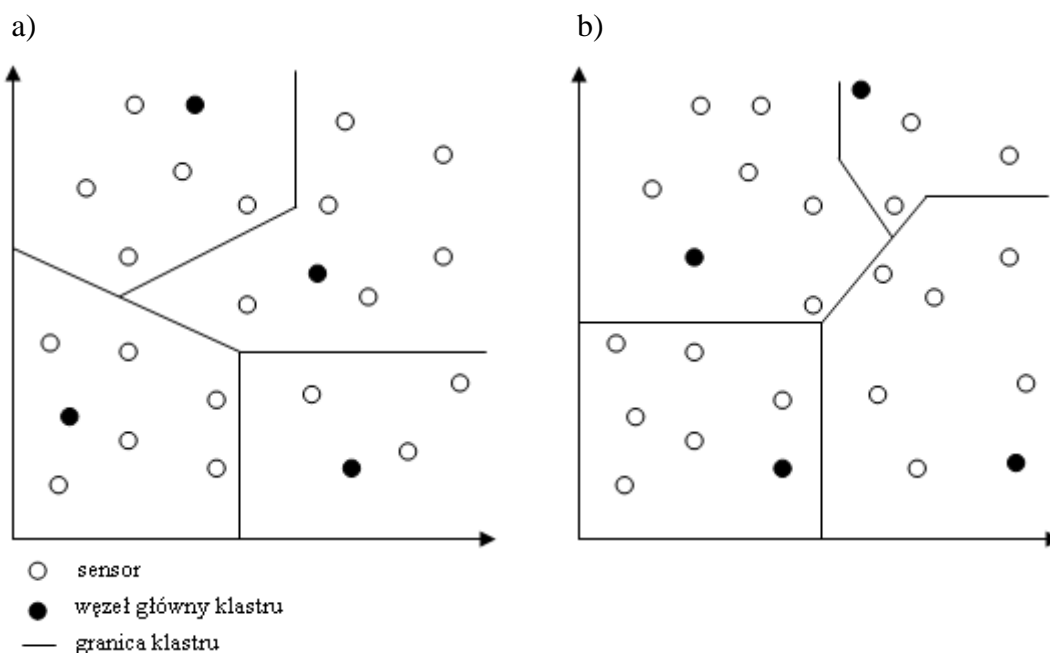
4.2. Hierarchiczne protokoły routingu

4.2.1. Low-Energy Adaptive Clustering Hierarchy

Protokół Low-Energy Adaptive Clustering Hierarchy (LEACH) dokonuje podziału sensorów na tzw. klastry (rys.13), następnie z grupy węzłów tworzących klaster wybiera węzeł nadrzędny, który pobiera i wysyła dane pomiarowe do bramy sieci sensorowej. Komunikacja węzła nadrzędnego danego klastra z bramą (stacją bazową) całej sieci pochłania wiele energii ze względu na dość dużą odległość, dlatego też protokół podzielono na tzw. rundy. Zastosowany podział umożliwia równomierne zużycie energii w poszczególnych sensorach tworzących klaster, gdyż w każdej kolejnej rundzie rolę węzła nadrzędnego (głównego) pełni inny sensor. Każda runda przebiega w dwóch fazach:

- setup,
- steady.

W fazie *setup* (zestawiania) tworzone są klastry oraz wybierane sensory pełniące funkcję węzła nadrzędnego (głównego), natomiast w fazie *steady* (ustalonej) jest realizowana transmisja danych pomiarowych.



Rys. 13. Podział sensorów na klastry w kolejnych dwóch różnych rundach: a) runda n, b) runda n+1 [10]

Na rys.13 widzimy, że w kolejnych rundach zmienia się nie tylko sensor pełniący rolę węzła głównego, ale także granice klastru.

W początkowej fazie organizacji klastrów (faza *steup*) wybierany jest sensor pełniący funkcję węzła głównego. Sensory same podejmują decyzję o tym czy chcą pełnić funkcję węzła głównego po przez losowy wybór liczby z przedziału od 0 do 1. Wylosowana liczba jest porównywana z wartością progową, która określana jest na podstawie zależności[10]:

$$T(n) = \frac{P}{1 - P \times \left(r \bmod \frac{1}{P} \right)} \quad \text{jeśli } n \in G \quad (1)$$

$$T(n) = 0 \quad \text{jeśli } n \notin G$$

gdzie: P – procentowa chęć objęcia przez sensory funkcji węzła głównego,

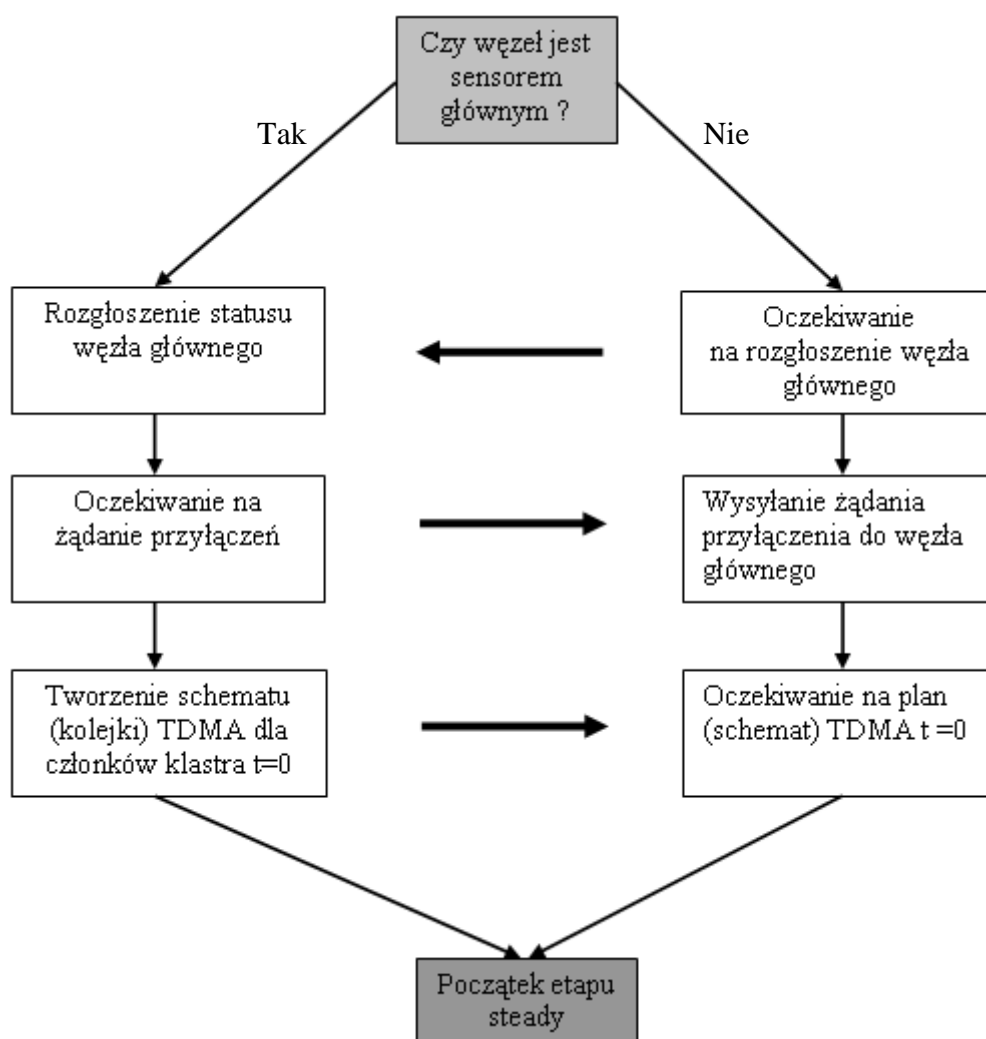
r – numer rundy,

G – zbiór sensorów, które nie pełniły funkcji węzła głównego klastru przez $\frac{1}{P}$ rund.

Gdy sensor wylosował liczbę mniejszą od wartości $T(n)$ określonej zależnością (1) to wówczas staje się węzłem głównym klastru. Taki sposób wyboru węzła głównego gwarantuje, iż każdy sensor będzie pełnił tą funkcję tylko raz w trakcie $1/P$ rund. Sensor,

który już raz został węzłem głównym, np. w rundzie $r = 2$ nie jest brany pod uwagę w kolejnych rundach (usuwany ze zbioru G), a zatem prawdopodobieństwo pełnienia funkcji węzła głównego w kolejnych rundach wzrasta i po upływie $1/P$ rund powraca do stanu początkowego (w tym momencie każdy sensor ma równe szanse zostać węzłem głównym).

Po wybraniu wszystkich węzłów głównych w klastrach, każdy z nich rozsyła informację, że w danej rundzie to on będzie pełnił taką rolę. Rozsyłana wiadomość składa się z ID sensora oraz nagłówka (określa informację broadcast). Sensory będące zwykłymi węzłami po odbiorze takiej informacji podejmują decyzję do jakiego węzła głównego przyłączyć się (na ogół ten wybór uzależniony jest od jakości odbieranego sygnału). Po dokonaniu wyboru sensory wysyłają wiadomość wyrażającą chęć przyłączenia się do danego węzła głównego (przesyłany pakiet zawiera ID sensora oraz węzła głównego). W ten sposób organizowany jest klaster, którego granice ustalane są tylko na jedną rundę. Przebieg całej fazy wybierania węzłów głównych oraz tworzenia klastrów przedstawiono na rys. 14.



Rys. 14. Ilustracja przebiegu fazy zestawiania (setup) w protokole LEACH[10]

Etap *setup* kończy proces kolejkwania TDMA, w którym węzeł główny danego klastru określa czasy transmisji pozostałych sensorów, które wyraziły chęć przynależenia do niego. Następnie w celu uniknięcia kolizji w transmisji poszczególnych sensorów węzeł główny rozsyła do nich informację o przydzielonych czasach. Po ustaleniu schematu kolejkwania TDMA faza *setup* zostaje zakończona.

W fazie *steady* sensory realizują transmisję danych pomiarowych do węzła głównego w stałych przedziałach czasowych, których wielkości zależy głównie od liczby sensorów w klastrze. Węzeł główny po odebraniu danych od wszystkich sensorów dokonuje ich agregacji aby uniknąć nakładania (dublowania) się tych samych informacji.

W zorganizowanej przez protokół LEACH strukturze sieci może pojawić się problem zakłócania się dwóch różnych klastrów. Aby to niekorzystne zjawisko ograniczyć zastosowano modulowanie nośnej sekwencją unikalnego kodu, który jest inny w każdym klastrze (sensory działające w granicach danego klastra stosują te same kody).

Istnieje również pewna modyfikacja protokołu LEACH tzw. LEACH - C (ang. *Centralized*), w którym nieco inaczej realizowany jest wybór węzła głównego klastru, a mianowicie w trakcie etapu tzw. zestawiania (faza ustalona jest identyczna jak w LEACH) każdy z sensorów transmituje do tzw. bramy informację o swoim położeniu, którą uzyskuje z odbiornika GPS (oznacza to konieczność wbudowania odbiorników GPS w sensorach). Po otrzymaniu informacji o pozycjach poszczególnych sensorów oraz ich poziomie energii, brama czyli tzw. węzeł nadrzędny całej sieci sensorowej sam wybiera na podstawie informacji o zasobie energii węzła głównego klastru.

W tym protokole czas działania sieci sensorowej można przedłużyć obniżając wartość progową $T(n)$, która w modyfikacji LEACH – C określana jest zależnością:

$$T(n) = \frac{P}{1 - P \times \left(r \bmod \frac{1}{P} \right)} \cdot \frac{E_{n_current}}{E_{n_max}} \quad (2)$$

gdzie: $E_{n_current}$ - aktualny (obecny) poziom energii,

E_{n_max} - maksymalny poziom energii.

Zwiększenie żywotności sieci ma jednak swoje ujemne strony, a mianowicie po przeprowadzeniu każdej kolejnej rundy zasoby energetyczne sensora maleją, a więc automatycznie maleje wartość progowa $T(n)$. Ponadto zapewne będzie występowała sytuacja, w której pozostałe sensory mające odpowiednią ilość energii do porozumiewania się

z węzłem nadrzędnym całej sieci sensorowej nie będą w stanie określić liczby mniejszej od progu wyznaczonego zależnością (2).

4.2.2. Power-Efficient Gathering in Sensor Information Systems

Protokół Power-Efficient Gathering in Sensor Information Systems (PEGASIS) [11] stanowi rozszerzenie algorytmu LEACH. W tym protokole podobnie jak w LEACH sensory pełniące funkcje węzła głównego w klastrze wybierane są rundowo (jeśli komunikacja wszystkich węzłów głównych z bramą sieci zakończy się to rozpoczyna się kolejna runda w której następuje wybieranie nowych sensorów do komunikacji z bramą).

Główna modyfikacja tego protokołu w stosunku do LEACH polega na tym, iż wybrane sensory łączą się tylko ze swoimi najbliższymi sąsiadami. Określony sensor poszukuje najbliższego sąsiada mierząc moc sygnału na podstawie której wyznacza odległości. Po zlokalizowaniu najbliższego sąsiada dostraja swoją moc nadawania w taki sposób aby jego sygnał odbierany był tylko i wyłącznie przez jeden sensor (najbliżej położony). W ten sposób tworzona jest najkrótsza ścieżka do bramy nadzorującej całą sieć sensorową. Reasumując tworzony jest tzw. łańcuch w którym tylko jeden sensor realizuje transmisję danych pomiarowych do bazy całej sieci.

Naturalnie takie rozwiązanie przedłuża żywotność całej sieci poprzez równomierne zużycie energii poszczególnych sensorów, ale niestety ma to i ujemne strony, gdyż najprawdopodobniej wszystkie węzły zakończą swoją działalność w tym samym czasie.

4.2.3. Threshold sensitive Energy Efficient sensor Network

Protokół TEEN (*ang. Threshold sensitive Energy Efficient sensor Network*) [11] podobnie jak PEGASIS bazuje na rozwiązaniach przyjętych w algorytmie LEACH, czyli również wybierany jest sensor główny oraz tworzone są klastry. W tym protokole sensor główny przesyła do pozostałych węzłów (sensorów) w klastrze dwie wartości progowe:

- hard threshold (twardą),
- soft threshold (miękką).

Ponadto przydziela im szczeliny czasowe w technice TDMA.

Pierwszy z wymienionych progów określa minimalną wartości atrybutu przy którym sensory z obrębu danego klastru będą wykonywały i przysyłały pomiary do sensora głównego, natomiast drugi określa minimalny zakres zmian tego samego atrybutu powyżej

progu twardego przy którym sensor ponownie rozpocznie transmisję. Przyjęte rozwiązanie z wartościami progowymi ma swoje zalety jak i wady, a mianowicie nie wątpliwą zaletą jest ograniczenie liczby transmisji, natomiast wadą jest brak danych od określonego sensora jeśli przyjęte progi nie zostaną przekroczone (komunikacja wówczas nigdy nie nastąpi).

4.2.4. Adaptive Threshold sensitive Energy Efficient sensor Network

Protokół Adaptive Threshold sensitive Energy Efficient sensor Network (APTEEN) stanowi rozszerzenie omówionego wyżej protokołu TEEN. Podobnie jak w TEEN stosowane są dwie wartości progowe: twarda i miękka oraz przydzielana jest szczelina czasowa w technice TDMA. Protokół APTEEN różni się od TEEN tym iż posiada tzw. *count time* (CT) wyznaczający maksymalny czas pomiędzy kolejnymi partiami danych pomiarowych transmitowanych do bramy. Ponadto w APTEEN wprowadzono trzy rodzaje zapytań umożliwiających min:

- kontrolowanie w przyszłości danego zjawiska(zdarzenia) przez z góry określony czas,
- analizowanie bieżącego stanu sieci,
- przetwarzanie danych pomiarowych wykonanych w przeszłości.

4.3. Lokalizacyjne protokoły routingu

Sensory w lokalizacyjnych protokołach routingu adresowane są w zależności od swojej lokalizacji (położenia). Sensory mogą wyznaczać swoje położenie samodzielnie na podstawie sygnałów odbieranych z bazy, o czym będzie mowa nieco później, bądź czerpać tę informację z odbiorników GPS. Znajomość położenia umożliwia wyznaczenie odległości między sensorami, a tym samym określić najbardziej korzystną pod względem kosztów trasę do transmisji pakietów pomiarowych pomiędzy danym sensorem a bramą.

Wśród tej grupy protokołów możemy wyróżnić np. GAF[12], w którym tzw. pole sensorowe (teren na którym zostały rozmieszczone sensory) dzielone jest na obszary. Przynależność danego sensora do określonego obszaru uzależniona jest od jego lokalizacji. W obrębie wyznaczonego obszaru każdy z sensorów ma przypisany taki sam koszt transmisji danych. Następnie z każdego obszaru wybierany jest sensor, który będzie przysyłał dane pomiarowe pochodzące od sąsiadów do bazy. Taka organizacja sieci umożliwia zmniejszenie zużycia energii.

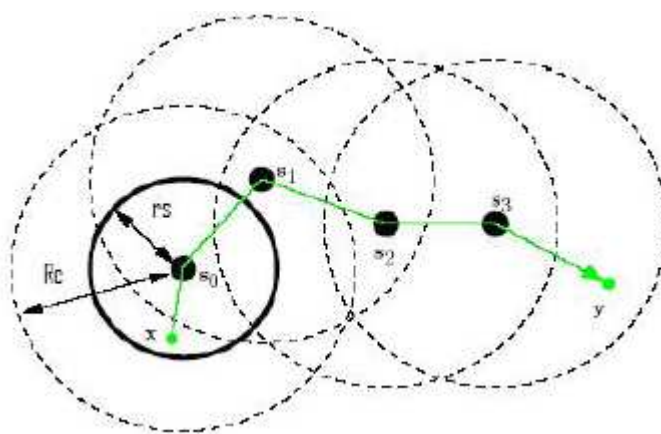
W protokole GAF można wyróżnić trzy fazy:

- wyszukiwanie sąsiadów zlokalizowanych w tym samym obszarze,
- uczestniczenie sensorów w trasowaniu danych pomiarowych,
- wyłączenie modułu radiowego.

5. Wykorzystanie technologii GIS w projektowaniu sieci sensorowych

Projektowanie sieci sensorowych wiąże się jak to już zostało powiedziane na wstępie z odpowiednim rozmieszczeniem czujników pod względem łączności rozumianej jako zdolność do komunikacji pomiędzy wszystkimi sensorami oraz pokrycia czyli możliwości monitorowania wyznaczonej uprzednio przestrzeni.

Zdolność monitorowania (pokrycie) oraz łączność w sieci sensorowej została przedstawiona na rys. 15.



Rys. 15. Pokrycie i łączność w sieci sensorowej (x, y -zdarzenia, R_c - zasięg radiowy, S_i -węzły sensorowe, r_s - czułość sensora, zdolność monitorowania określonego parametru)

Łączność w sieci sensorowej uzależniona jest od istnienia ścieżki pomiędzy dwoma węzłami, która może ulegać ciągłym zmianom na skutek mobilność węzłów ich uszkodzeń oraz ataków. Konsekwencją takich zdarzeń może być utrata ścieżki zapewniającej łączność, a przez to izolacja niektórych węzłów oraz chwilowe rozdrobnienie sieci do momentu generacji nowej ścieżki. W idealnym przypadku łączność w sieci WSN możemy modelować w postaci grafu $G(V, E)$ gdzie V jest zbiorem wierzchołków (węzłów), a E jest zbiorem krawędzi (ścieżek). Taki graf można nazwać k – połączeniowym jeżeli istnieje w nim przynajmniej k rozłącznych ścieżek pomiędzy każdą parą węzłów (sensorów) $u, v \in V$. Przyjmując tą analogię zdolność do realizacji połączenia można zapisać w postaci[13]:

$$u(R) = \frac{N\pi R^2}{A} \quad (3)$$

gdzie: R - promień transmisji (zasięg), A - powierzchnia, N - numer węzła na powierzchni A

W praktyce środowisko w którym będzie realizowana transmisja ma bardzo istotny wpływ na zasięg radiowy sensorów (nawiązywanie łączności). Na sygnał pomiędzy nadajnikiem a odbiornikiem poszczególnych sensorów mają wpływ różne przeszkody naturalne lub wytworzone przez człowieka wywołujące takie zjawiska jak dyfrakcja, rozpraszanie, odbijanie, które wpływają na opóźnienie, drogę propagacji sygnału pomiędzy Tx a Rx, a ponadto powodują jego fluktuację oraz tworzenie się obszarów zacienionych (tzw. efekt cienia radiowego, którego nasilenie w dużym stopniu zależy od częstotliwości) Zjawisko cienia radiowego (ang. shadowing) i tłumienia ścieżki (ang. path loss) są głównymi zagadnieniami w projektowaniu bezprzewodowych sieci sensorowych.

Tłumienie ścieżki można opisać modelem empirycznym opracowanym w pracy [14], którego formuła ma postać:

$$PL(d)[dB] = \overline{PL}(d) + X_{\sigma} = \overline{PL}(d_o) + 10\gamma \log\left(\frac{d}{d_o}\right) + X_{\sigma} \quad (4)$$

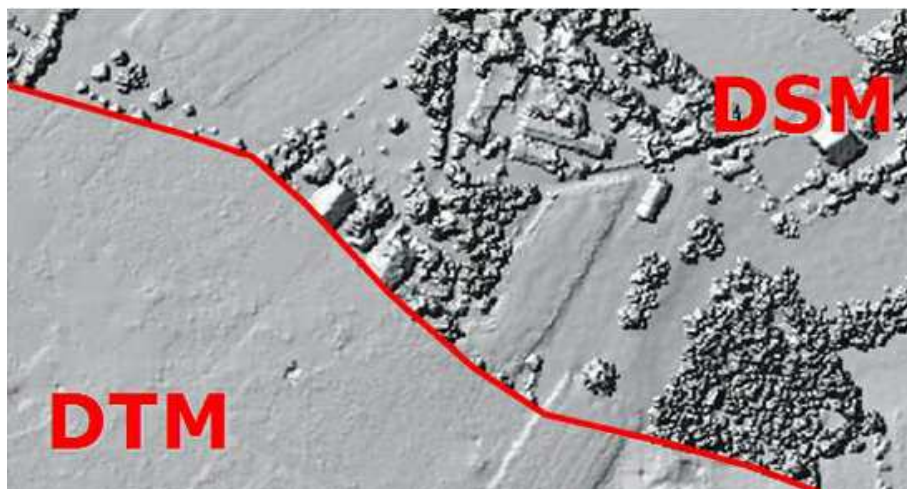
gdzie: X_{σ} - zerowa średnia rozkładu losowego Gaussa z odchyleniem standardowym σ .

γ - wykładnik tłumienia ścieżki.

Podany w zależności (4) model bierze pod uwagę praktyczne pomiary pokazujące fakt, że wartości średnia sygnału pomiędzy Tx a Rx znacznie różni się od wartości przewidywanej. Pomiary wykazały, że wartość tłumienia ścieżki na każdym dystansie d jest logarytmicznie normalna. Model ten opisuje również efekt zacieniania występujący przy dużej liczbie miejsc pomiarowych, które charakteryzują się taką samą odległością między Tx a Rx ale różnymi przeszkodami fizycznymi.

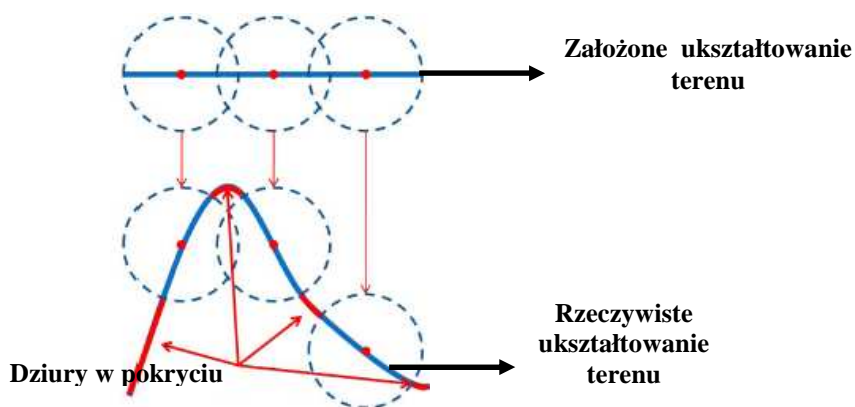
Warto tutaj zaznaczyć, że technologię GIS można wykorzystać w projektowaniu sieci WSN zarówno w obszarze łączności (zasięgu radiowego węzła) jak też pokrycia (zdolności do monitorowania). W przypadku predykcji zasięgu radiowego węzła (sensora) niezmiernie użytecznym module systemu GIS jest mapa cyfrowa NMT (rys.16), która dostarcza oprócz niezbędnych dla modelu propagacyjnego (zaimplementowanego w aplikacji wspierającej planowanie radiowe i posiadającej moduł GIS) danych obliczeniowych, także niezbędne informacje o typie środowiska (ukształtowaniu, lokalizacji różnych przeszkód). Posiadając odpowiednie oprogramowanie możemy wykonywać symulację zasięgów sensorów na wyznaczonym terenie ich potencjalnego działania. Uzyskane w trakcie symulacji mapy zasięgów pozwalają wizualnie ocenić pokrycie sygnałem radiowym przy założonych uprzednio parametrach wejściowych (moc, częstotliwość). Wykonywanie analiz predykcyjnych pozwala efektywnie rozmieszczać sensory, wyznaczać profil trasy

propagacyjnej oraz identyfikować miejsca w których może wystąpić zjawisko cienia radiowego.



Rys. 16. Numeryczny model terenu(DTM – ang. *digital terrain model*, DEM – ang. *digital elevation model*)

Tego typu analizy są niezwykle ważne w przypadku planowania sieci WSN pod względem czułości sensora (jego zasięgu monitorującego). Otóż jeśli rozmieścimy sensory równomiernie (w taki sposób aby ich zasięgi czułości pokrywały się w nie wielkim stopniu, a przy tym uzyskamy pełne pokrycie monitorujące danego obszaru) to z pewnością będzie mniejszy problem z powtarzającą się informacją w sieć – będzie mniejsze obciążenie sieci, ale może pojawić się problem z łącznością, której brak na skutek różnych przeszkód terenowych (obszarów cieniowanych) może prowadzić w konsekwencji do izolacji niektórych sensorów (a tym samym do braku informacji i niepełnego pokrycia). Ponadto brak informacji o ukształtowaniu terenu w trakcie planowania pokrycia monitorującego może w konsekwencji prowadzić do sytuacji zobrazonej na rys.17.



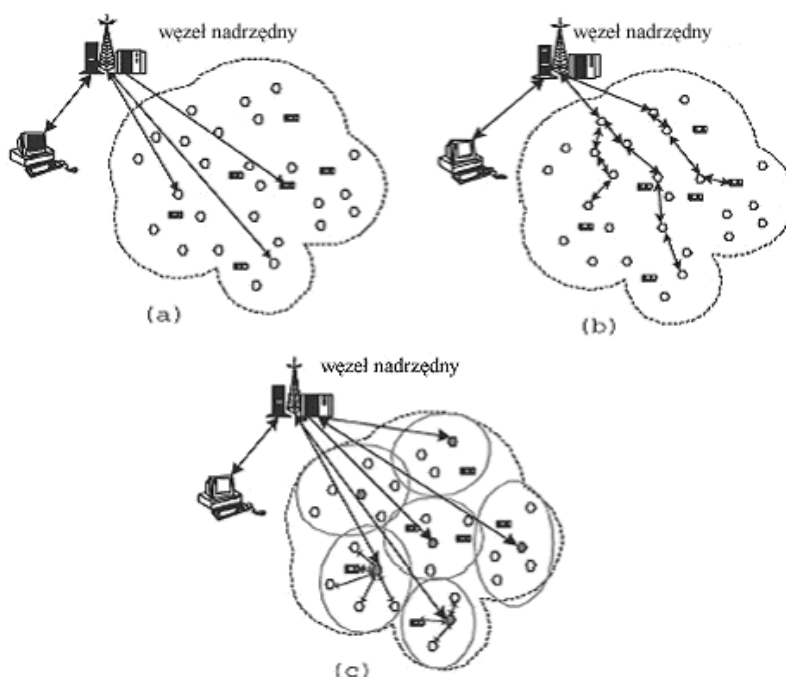
Rys. 17. Ilustracja dziur w pokryciu (czułości monitorującej) sieci WSN w przypadku braku informacji o ukształtowaniu terenu

Oczywiście w przypadku rozrzucania sensorów na dany teren z samolotu trudno tu mówić o racjonalnym ich rozmieszczeniu zarówno pod względem łączności jak też czułości monitorowanego parametru, ale po uzyskaniu informacji o położeniu sensorów (dzięki odpowiednim algorytmom lokalizacyjnym) możemy zobrazować powstałą strukturę sieci w oprogramowaniu GIS, a tym samym zarządzać i optymalizować, o czym będzie nieco więcej w kolejnych rozdziałach.

6. Wyznaczanie położenia węzłów sieci sensorowej

Węzy (sensory) sieci sensorowej zazwyczaj rozmieszczane są w sposób przypadkowy i mogą pracować w trzech strukturach (rys.18):

- a) jednolitej;
- b) hierarchicznej;
- c) bezpośredniej.



Rys. 18. Struktury sieci sensorowych: a) bezpośrednia, b) jednolita, c) hierarchiczna [2]

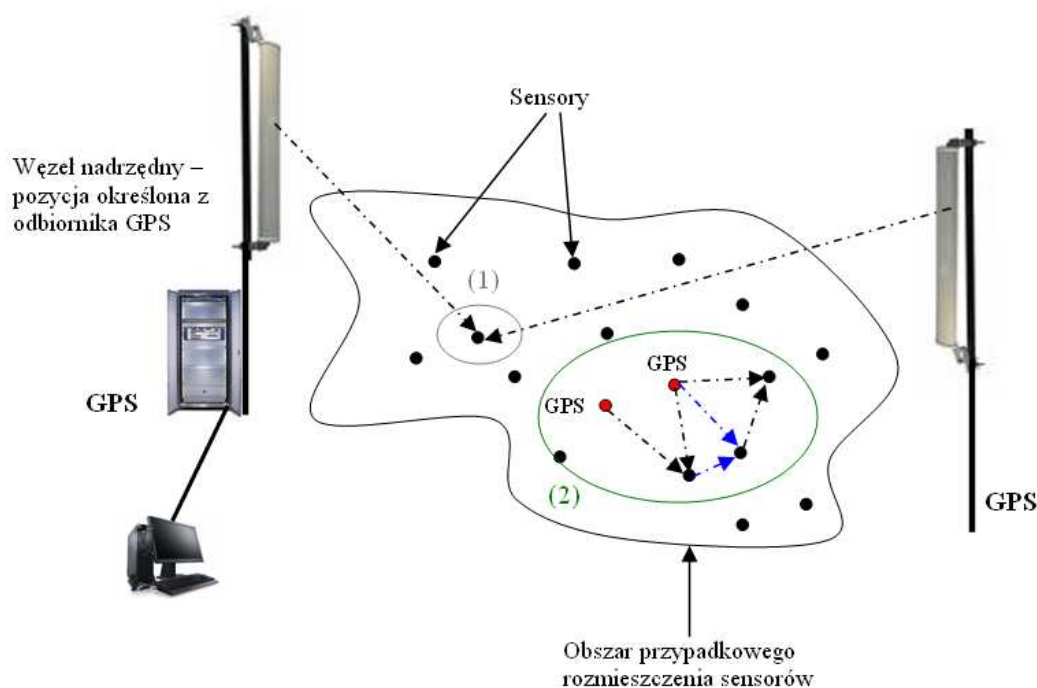
Spośród wymienionych architektur sieci największe znaczenie ma struktura jednolita i hierarchiczna.

Aby sensory po zorganizowaniu się w sieć były w stanie samodzielnie wyznaczyć swoje położenie na podstawie pozycji przynajmniej dwóch sąsiednich węzłów konieczna jest

znajomość ich współrzędnych. Jeśli założymy że współrzędne mogą być odczytane za pomocą odbiorników GPS to można tutaj rozpatrywać dwie sytuacje które ilustruje rys.19.

W pierwszym wariancie (1) można założyć, że cały obszar na którym są rozmieszczone sensory jest w zasięgu dwóch węzłów nadrzędnych (tzw. bram), których pozycje są znane na podstawie współrzędnych GPS. W tej sytuacji każdy z sensorów może na podstawie odebranych sygnałów z bram obliczyć swoją pozycje (dzięki odpowiednim algorytmom zaimplementowanym w mikrokontrolerze sensora). Ten wariant poczęci bazuje na strukturze sieci bezpośredniej to znaczy, że rozkazy pomiaru określonego parametru mogłyby być kierowane do danego sensora bezpośrednio z węzła nadrzędnego, natomiast wyniki zwrótne byłyby przesyłane między poszczególnymi sensorami i dopiero najbliższy z nich nawiązywał by połączenie z węzłem nadrzędnym.

W drugim przypadku (2) wśród rozmieszczonych sensorów muszą być przynajmniej dwa z wbudowanymi odbiornikami GPS oraz znajdować się stosunkowo blisko siebie. Dany sensor, który odbierze sygnały z węzłów o określonej pozycji będzie w stanie obliczyć swoje współrzędne (na podstawie opracowanych algorytmów). Później ten proces może być kontynuowany na zasadzie pętli to znaczy, że każdy kolejny sensor będzie mógł określić swoją pozycje na podstawie sygnałów odebranych z innych sensorów, które wyznaczyły już swoje położenie bazując na innych sensorach (proces ten rozpoczyna się od dwóch węzłów z odbiornikami GPS).



Rys.19. Ilustracja wyznaczania pozycji sensorów na podstawie odbieranych sygnałów z węzłów nadrzędnych (1) oraz sensorów o znanych współrzędnych (2)

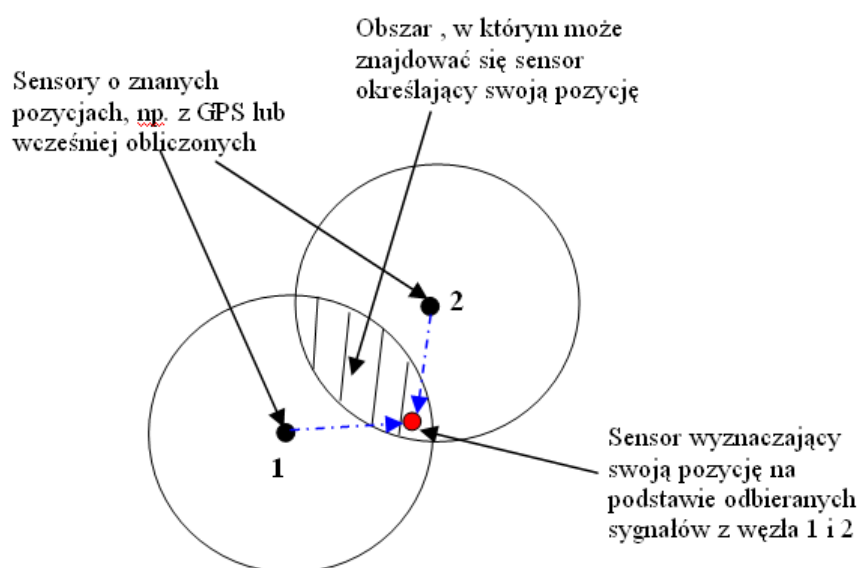
Zarówno w pierwszym jak i drugim przypadku mamy do czynienia z sytuacją przedstawioną na rys.20.

Sygnaly odbierane przez sensor (rys.20) z węzłów 1 i 2 na skutek odbić od przeszkód terenowych mają różne fazy oraz amplitudy. Na podstawie funkcji korelacji tych sygnałów możemy określić ich wzajemne opóźnienia oraz długość dróg jakie przebyły (odległość sensora od węzła 1 i 2). Obliczoną odległość możemy wyrazić w postaci zależności na długość wektora.

Oznaczmy sensor (rys.20) wyznaczający swoją pozycję literą **Z** wówczas jego odległość od węzłów 1 i 2 możemy zapisać w postaci układu równań:

$$\begin{cases} |\vec{1Z}| = \sqrt{(X_Z - X_1)^2 + (Y_Z - Y_1)^2} \\ |\vec{2Z}| = \sqrt{(X_Z - X_2)^2 + (Y_Z - Y_2)^2} \end{cases} \quad (5)$$

W układzie (1.1) niewiadomymi są X_Z i Y_Z .



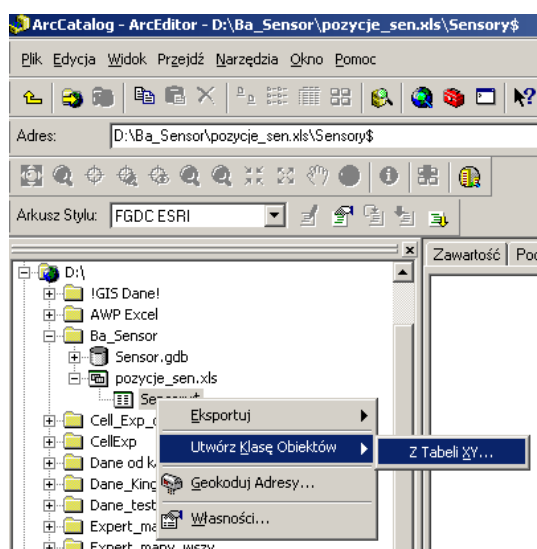
Rys. 20. Ilustracja obszaru w obrębie którego może znajdować się sensor wyznaczający swoją pozycję – jest to sytuacja bardzo wyidealizowana

Wobec powyższego w mikrokontrolerze sensora musi być zaimplementowany odpowiedni algorytm obliczający funkcje korelacji i rozwiązujący układ równań (5). Na podstawie tych algorytmów sensor określi swoje współrzędne w układzie odosobnionym i prześle je do węzła nadrzędnego.

7. Zastosowanie systemu GIS w lokalizacji węzłów sieci sensorowej

Węzły sieci sensorowej dzięki zaimplementowaniu algorytmów opierających się na funkcji korelacji oraz układzie równań (5) mogą wyznaczyć swoje położenie w odosobnionym układzie współrzędnych. Dokonując transformacji tego układu na odpowiedni układ geograficzny można wyznaczone pozycje sensorów rozmieścić na mapie. Transformację układów można przeprowadzić wykorzystując np. pakiet oprogramowania ArcGIS (ArcMap, ArcCatalog, ArcToolBox) firmy ESRI.

Korzystając z aplikacji ArcCatalog, która umożliwia organizowanie, zarządzanie i przetwarzanie danych można utworzyć z wyliczonych pozycji sensorów plik „Shape”. Plik ten będzie warstwą tematyczną mapy w aplikacji ArcMap, która zawierać będzie informację o rozmieszczeniu sensorów na danym obszarze. Utworzenie pliku „Shape” wymaga posiadania danych w odpowiednim formacie. Dane mogą mieć postać tabeli (np. Excel, bądź pochodzić z relacyjnej bazy danych), w której poszczególne kolumny będą zawierały współrzędne danego sensora w układzie odosobnionym. Z danych w postaci tabeli można utworzyć plik „Shape” korzystając z funkcji *Utwórz Klasę Obiektów z tabeli XY* (rys. 21)



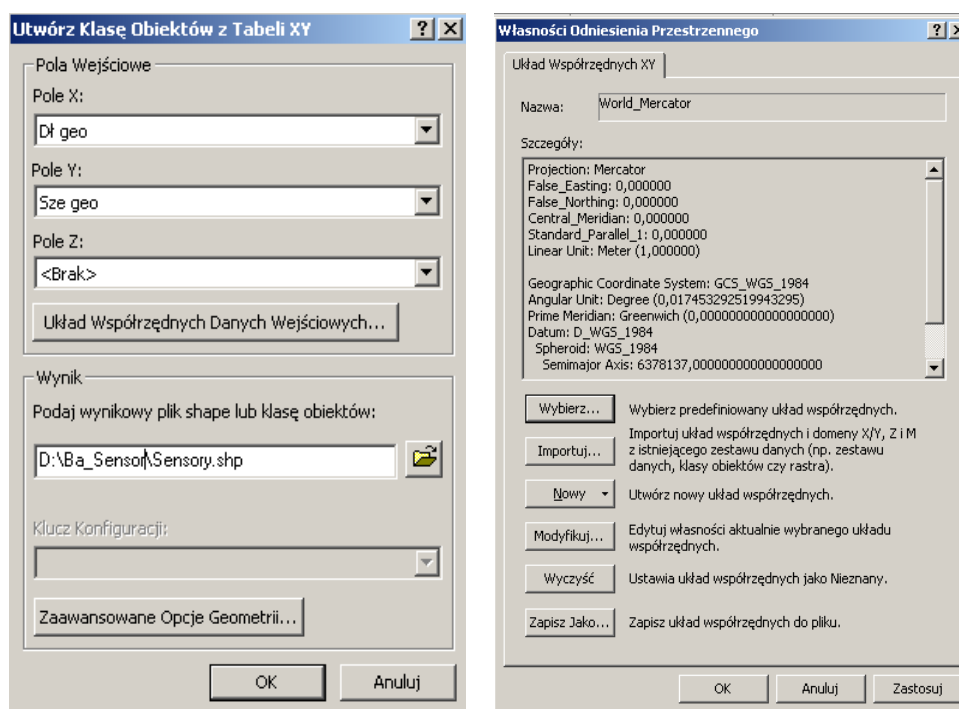
Rys. 21. Tworzenie pliku „Shape” w aplikacji ArcCatalog

W trakcie tworzenia pliku „Shape” dokonywana jest również transformacja układu odniesienia, w którym zostały wyznaczone współrzędne sensorów na odpowiedni układ współrzędnych geograficznych. Wybrany układ współrzędnych geograficznych musi być zgodny z układem, w którym są zapisane warstwy tematyczne mapy zawierające informacje o

danym obszarze np. lasy, budynki, zbiorniki wodne. Niezgodność układów stwarza problemy w trakcie wyświetlania poszczególnych warstw tematycznych mapy w aplikacji ArcMap.

Reasumując należy wybrać odpowiednie kolumny z tabeli, z których będzie tworzony plik „Shape” (w tym przypadku będzie to długość i szerokość geograficzna) oraz odpowiedni układ współrzędnych wejściowych (układ odosobniony) i wyjściowych (układ pożądaný) – rys.22. Pożądaný układ współrzędnych można wybrać z pośród istniejących w oprogramowaniu bądź utworzyć własny modyfikując już istniejące.

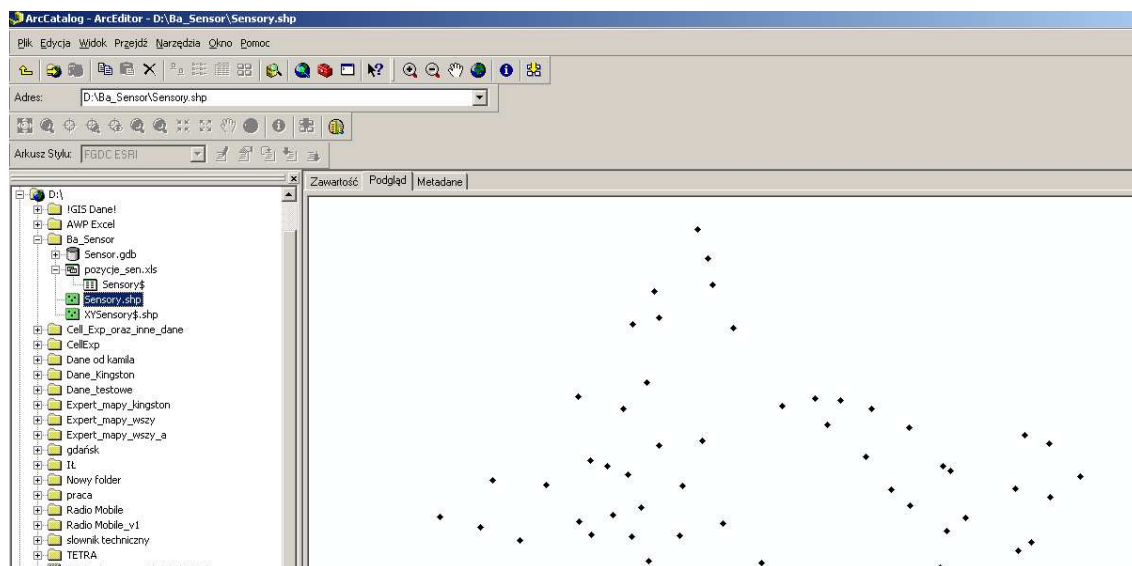
Po odpowiedniej konfiguracji wszelkich parametrów (patrz rys.22) aplikacja ArcCatalog na podstawie długości i szerokości geograficznej automatycznie rozmieści pozycje sensorów w pożądanym układzie współrzędnych, które będą zapisane w postaci pliku „Shape”- rys.23.



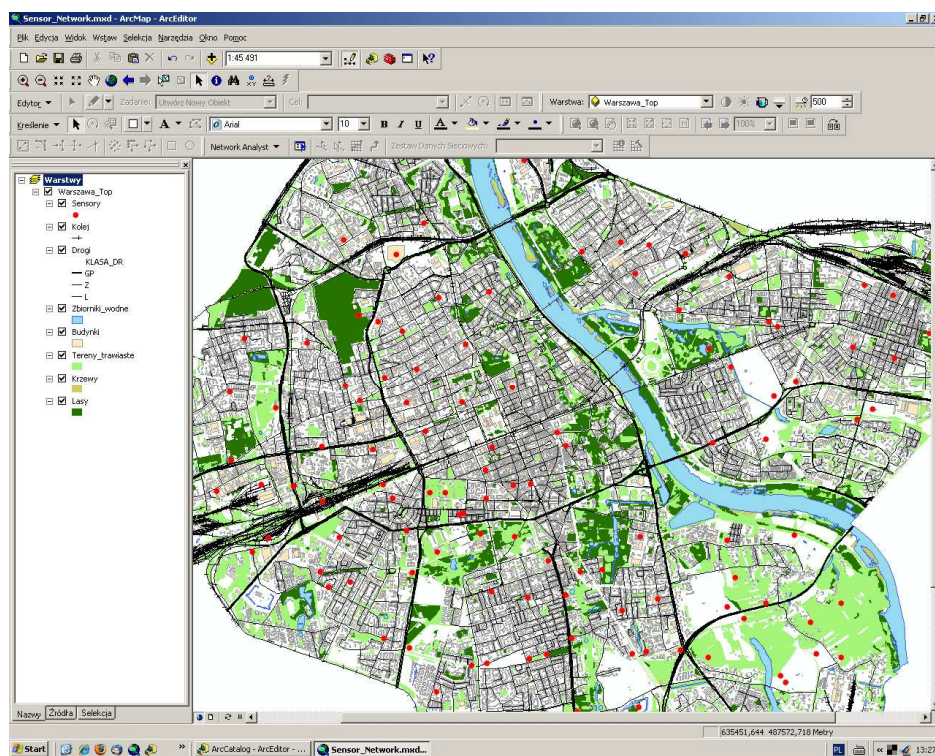
Rys. 22. Tworzenie pliku „Shape” w aplikacji ArcCatalog – wybór pól wejściowych (długości i szerokości geograficznej) oraz układów współrzędnych

Utworzony plik „Shape” stanowi jak już to zostało wspomniane w aplikacji ArcMap warstwę tematyczną (rys.24).

Należy tutaj podkreślić, że plik ten został wygenerowany na podstawie danych teoretycznych (w tabeli przyjęto teoretyczne pozycje sensorów w układzie odosobnionym). W rzeczywistości sensory po obliczeniu swojej pozycji przesyłają ją do węzła nadrzędnego, który dzięki dostępowi do sieci Internet zapisuje je np. w relacyjnej bazie danych. Z tej bazy dane mogą być pobrane przez system GIS.



Rys. 23. Rozmieszczone pozycje sensorów w przyjętym układzie współrzędnych

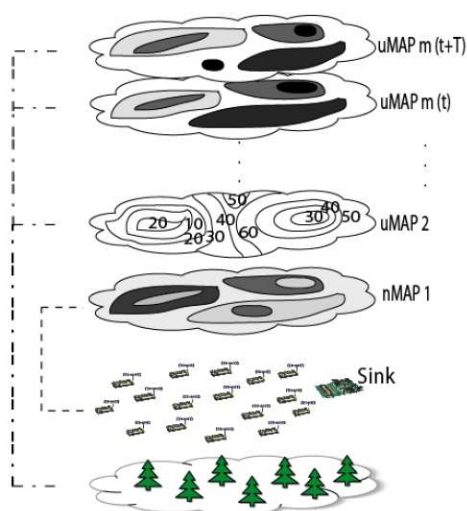


Rys. 24. Przykładowa mapa wykonana w aplikacji ArcMap – obrazuje rozmieszczenie sensorów na danym obszarze

Wykonana w aplikacji ArcGIS mapa (rys.24) obrazuje rozmieszczenie sensorów (kolor czerwony) na tle innych warstw tematycznych. Każda z warstw zawiera odrębną informację dotyczącą np. rozmieszczenia obszarów leśnych, budowli, zbiorników wodnych, itd.

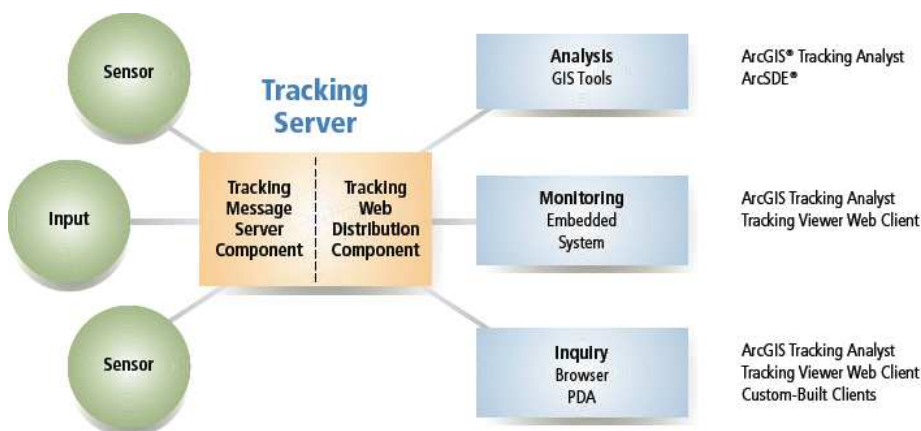
Dysponując taką mapą możemy wizualnie określić gdzie dany sensor się znajduje i z jakiego obszaru przesyła wyniki pomiarów oraz czy dany obszar jest w pełni monitorowany (czy sensory swoją czułością, zdolnością monitorującą pokrywają cały obszar – jest to

szczególnie istotne w przypadku rozrzucenia sensorów z samolotu, wówczas możemy wizualnie określić jakie jest pokrycie przy założonej czułości sensorów i ewentualnie podejmować decyzję o wyłączeniu niektórych sensorów z powodu pokrywania się w dużym procencie ich obszarów monitorowania). Dzięki temu możemy wykonywać różne analizy, np. określać obszar w obrębie, którego będzie występowała taka sama wartość mierzonego parametru (temperatury, wilgotność czy też stężenia szkodliwej substancji – w zależności od przeznaczenia danej sieci sensorowej) - jednym słowem istnieje możliwość tworzenia map prezentujących rozkład mierzonego parametru (rys.25).



Rys. 25. Idea tworzenia map rozkładu mierzonych parametrów przez sieć sensorową w oparciu o system GIS

Pomiary zbierane z danego obszaru przez sieć sensorową możemy wprowadzać do systemu GIS w czasie rzeczywistym wykorzystując np. aplikację Tracking Server firmy Esri (rys.26), następnie wykorzystując ogromne możliwości analityczne systemu GIS można na bieżąco wykonywać różnego rodzaju analizy, mapy prezentujące wyniki pomiarów.



Rys. 26. Ilustracja przekazywania danych przez Tracking Server do poszczególnych aplikacji klienckich[15]

Spis rysunków

Rys. 1. Architektura węzła sensorowego.	5
Rys. 2. Przykłady sensorów: a) radar samochodowy, b) czujnik pomiaru cieczy w zbiornikach, c) miernik tempera tury linii przesyłowej 420 kV –TU Darmstadt.....	6
Rys. 3. Uproszczona architektura sieci sensorowej.	6
Rys. 4. Struktura (model) warstwowa sieci sensorowej [1]	7
Rys. 5. Podział protokołów routingu w sieciach sensorowych.	10
Rys. 6. Ilustracja działania protokołu flooding (zalewania).....	12
Rys. 7. Ilustracja wad protokołu flooding: a) Implozja, b) Overlap [4].....	12
Rys. 8. Ilustracja działania protokołu gossiping	13
Rys. 9. Działanie protokołu SPIN[4]	14
Rys. 10. Działanie protokołu Directed Diffusion: a) dystrybucja zapytania, b) tworzenie gradientów, c) transmisja danych [5]	15
Rys. 11. Działanie protokołu Rumor Routing.....	16
Rys. 12. Zestawianie ścieżki w protokole Energy Aware Routing	17
Rys. 13. Podział sensorów na klastry w kolejnych dwóch różnych rundach: a) runda n, b) runda n+1 [10].....	19
Rys. 14. Ilustracja przebiegu fazy zestawiania (setup) w protokole LEACH[10]	20
Rys. 15. Pokrycie i łączność w sieci sensorowej (x, y -zdarzenia, R_c - zasięg radiowy, S_i -węzły sensorowe, r_s -czułość sensora, zdolność monitorowania określonego parametru) ...	24
Rys. 16. Numeryczny model terenu(<i>DTM – ang. digital terrain model, DEM – ang. digital elevation model</i>)	26
Rys. 17. Ilustracja dziur w pokryciu (czułości monitorującej) sieci WSN w przypadku braku informacji o ukształtowaniu terenu	26
Rys. 18. Struktury sieci sensorowych: a) bezpośrednia, b) jednolita, c) hierarchiczna [2]	27
Rys. 19. Ilustracja wyznaczania pozycji sensorów na podstawie odbieranych sygnałów z węzłów nadrzędnych (1) oraz sensorów o znanych współrzędnych (2).....	28
Rys. 20. Ilustracja obszaru w obrębie którego może znajdować się sensor wyznaczający swoją pozycję – jest to sytuacja bardzo wyidealizowana.....	29
Rys. 21. Tworzenie pliku „Shape” w aplikacji ArcCatalog	30
Rys. 22. Tworzenie pliku „Shape” w aplikacji ArcCatalog – wybór pól wejściowych (długości i szerokości geograficznej) oraz układów współrzędnych.....	31
Rys. 23. Rozmieszczone pozycje sensorów w przyjętym układzie współrzędnych	32
Rys. 24. Przykładowa mapa wykonana w aplikacji ArcMap – obrazuje rozmieszczenie sensorów na danym obszarze	32
Rys. 25. Idea tworzenia map rozkładu mierzonych parametrów przez sieć sensorową w oparciu o system GIS	33
Rys. 26. Ilustracja przekazywania danych przez Tracking Server do poszczególnych aplikacji klienckich[15]	33

Bibliografia

- [1] Akyildiz I.F., Sankarasubramaniam W. Su, Y., Cayirci E.: „*Wireless sensor networks: A survey*”, *Computer Networks* , 2002.
- [2] Jeznach J., Niepsuj P.: „*Analiza protokołów routingu w sieciach sensorowych*” ,AGH, 2007.
- [3] Braginsky D., Estron D.: „*Rumor Routing Algorithm For Sensor Networks*”, *Proceedings of the first Workshop on Sensor Networks and Applications*, Atlanta, USA, 2002
- [4] Heinzelman W., Kulik J., Balakrishnan H.: „*Adaptive Protocols for Information Dissemination in Wireless Sensor Networks*”,MIT, Cambridge, 1999.
- [5] Intanagonwiwat, Govindan, Estrin, Heidemann, and Fabio Silva, "*Directed Diffusion for Wireless Sensor Networking*", ACM Mobicom 2000.
- [6] Braginsky D., Estron D.: „*Rumor Routing Algorithm For Sensor Networks*”, Atlanta, USA, 2002.
- [7] Al-Karaki J. N., Kamal A.E.: „*Routing Techniques in Wireless Sensor Networks*, *IEEE*, 2004.
- [8] Shah R. C., Rabaey J., „*Energy Aware Routing for Low Energy Ad Hoc Sensor Networks*", *IEEE Orlando*, USA, 2002.
- [9] Leuschner C. J.: „*The design of a simple energy efficient routing protocol to improve sensor network lifetime*”, *University of Pretoria*, 2005.
- [10] Heinzelman wireless B., Chandrakasan A., Balakrishnan H., „*An Application-Specific Protocol Architecture for Wireless Microsensor Networks*”, *IEEE* , 2002.
- [11] Manjeshwar A., Agrawal D. P.: „*TEEN : A Protocol for Enhanced Efficiency in Wireless Sensor Networks*”, *Proc. 1st International Workshop on Parallel and Distributed Computing Issues in Wireless Networks and Mobile Computing*, San Francisco, USA, 2001.
- [12] Y. Xu, J. Heidemann, D. Estrin, „*Geography-informed Energy Conservation for Ad- hoc Routing*,", 2001.
- [13] Benahmed Khelifa1, H. Haffaf , Merabti Madjid, David Llewellyn-Jones.: „*Monitoring Connectivity in Wireless Sensor Networks*”, *International Journal of Future Generation Communication and Networking* vol. 2, No. 2, June, 2009.
- [14] Tomáš Mácha, Peter Stančík, Vít Novotný.: „*Connectivity in a wireless sensor network*”, *IJCSNS International Journal of Computer Science and Network Security*, VOL.8 No.12, December 2008.
- [15] www.esri.com.