

Roboty monitorujące skażenie środowiska - symulator V-REP

Anna Wujek
Łukasz Korpala
Wiktor Ślęczka

10 czerwca 2016

Spis treści

1	Zadanie	3
2	Projekt systemu	3
2.1	Założenia	3
2.2	Scenariusz działania	3
2.3	Struktura systemu	4
2.3.1	Symulator przestrzeni roboczej	4
2.4	Wstępny projekt systemu	5
3	Realizacja	5
3.1	Komunikacja	5
3.1.1	Roboty	6
3.1.2	Chmura	6
3.2	Roboty	6
3.3	Algorytmy	6
3.3.1	Algorytm poszukiwania chmury	6
3.3.2	Algorytm bug 2	6
3.3.3	Algorytm otaczania chmury	8
3.4	Przykładowe mapy środowiska	9

1 Zadanie

Celem projektu było stworzenie symulatora mobilnej sieci ad hoc (MANET) do monitorowania skażenia środowiska naturalnego. Węzłami sieci są roboty mobilne, wyposażone w czujniki oraz komunikujące się między sobą. Zadaniem robotów jest lokalizacja chmury skażenia i otoczenie jej robotami, monitorującymi jej położenie i granice, przy założeniu utrzymania spójności sieci.

2 Projekt systemu

2.1 Założenia

- Roboty potrafią się na bieżąco lokalizować.
- System składa się z pewnej liczby robotów mobilnych.
- Roboty mobilne działają autonomicznie i potrafią same zorganizować sieć.
- Roboty są wyposażone w czujniki pozwalające im wykrywać poziom skażenia oraz przeszkody.
- Roboty są wyposażone w urządzenia pozwalające im się między sobą komunikować się między sobą.
- Chmura skażenia jest ogólnie określona i niezmienna w trakcie trwania symulacji.
- Nieznana jest mapa przestrzeni roboczej, ale znane są jej granice i kształt - prostokąt.

2.2 Scenariusz działania

Algorytm realizowany przez każdego robota:

1. Warunek początkowy: roboty znajdują się w dowolnym punkcie przestrzeni na mapie. Z punktu tego musi istnieć droga prowadząca do obszaru poszukiwań.
2. Przenieść się do punktu początkowego, zależnego od liczby robotów, wymiarów przestrzeni poszukiwań i zasięgu czujników, zgodnie ze wzorem:

$$max_spaces = sensor_range \cdot precision_range \quad (1)$$

$$x = -length/2 \quad (2)$$

$$y = width/2 + distances * robot_number + distances/2 \quad (3)$$

3. Wyznacz kolejny punkt docelowy, zgodnie ze wzorem:

$$fvdfv = vfdknvdk \quad (4)$$

4. Udaj się do punktu docelowego, monitorując skażenie oraz wykrywając przeszkody.

5. Jeśli wykryto przeszkodę, omiń ją algorytmem Bug2. Wyznacz następny punkt docelowy, jest aktualny został minięty.
6. Jeśli wykryto chmurę skażenia, zatrzymaj się.
7. Jeśli cały obszar poszukiwań sprawdzony, to koniec algorytmu; jeśli nie, to przejdź do punktu 3.

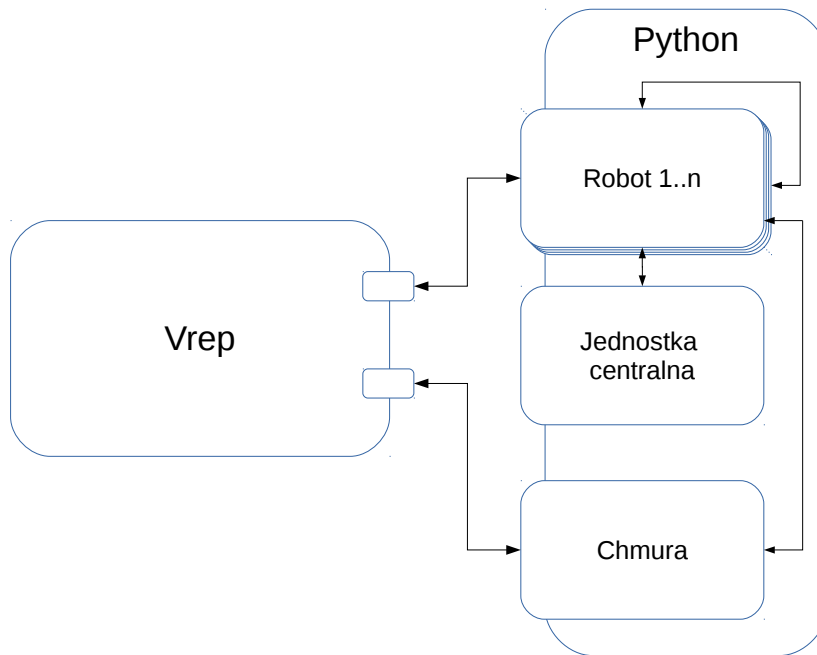
2.3 Struktura systemu

System prezentowany w projekcie składa się z następujących elementów:

- **symulator przestrzeni roboczej:**
odpowiedzialny za wizualizację oraz fizyczne oddziaływania pomiędzy elementami (np. kolizje), a także za wyznaczanie odczytów czujników (oprócz czujników skażenia);
- **symulator robotów:**
odpowiedzialny za wyznaczanie sterowania robotów na podstawie odczytów z czujników, realizujący określone zadanie;
- **symulator chmury skażenia:**
odpowiedzialny za generowanie odczytów czujników skażenia;
- **komunikacja:**
zapewnia przekazywanie informacji pomiędzy poszczególnymi symulatorami.

2.3.1 Symulator przestrzeni roboczej

2.4 Wstępny projekt systemu



Rysunek 1: Schemat działania systemu.

Dla robotów, jednostki centralnej oraz chmury stworzone zostaną procesy, które będą się komunikować z symulatorem V-rep:

- procesy robotów będą wysyłać sterowania i odbierać odczyty z czujników przeszkód z symulatora V-rep (V-rep będzie symulować rzeczywiste roboty i rzeczywiste środowisko);
- możliwa będzie komunikacja pomiędzy robotami;
- jednostka centralna będzie odbierać od robotów odczyty skażenia;
- stworzony zostanie osobny proces, który będzie symulował chmurę skażenia; proces ten symulował będzie wartości odczytów czujników skażenia i przekazywał informacje o kształcie i położeniu chmury do V-repa.

3 Realizacja

3.1 Komunikacja

Komunikacja przebiega na zasadzie ogłaszania aktualnych informacji na swój temat przez roboty (oraz chmurę). Informacje przesyłane są za pomocą struktury JSON do gniazd, przypisywanych każdemu modułowi osobno. Wiadomości opatrzone są nagłówkiem, pozwalającym sklasyfikować komunikat. Dane tworzone są w postaci usystematyzowanych słowników, których struktura została ogólnie ustalona. Komunikaty są wysyłane i odbierane asynchronicznie.

3.1.1 Roboty

3.1.2 Chmura

3.2 Roboty

Wykorzystywane w realizacji zadania roboty Pioneer 3-DX wyposażone w ultradźwiękowe czujniki odległości - będą one zapewnione przez symulator V-Rep. Posiadają one napęd różnicowy, pozwalający na dużą swobodę poruszania się po symulowanej przestrzeni. Są przystosowane do rozbudowy o dodatkowe urządzenia - dzięki czemu mogłyby być łatwo dostosowane do realizacji zadania lokalizacji chmury. Dodatkowo, czujniki stężenia toksycznych substancji będą realizowane w skryptach Pythona, ułatwiając jej symulację. Informacje o jej położeniu i kształcie będą przekazywane do Vrepa, gdzie nastąpi ich wizualizacja.

3.3 Algorytmy

3.3.1 Algorytm poszukiwania chmury

Roboty rozpoczynają pracę od ustawienia się w formacji linii - maksymalna odległość między robotami jest określona wzorem: $d_{max} = 0.7 * \min(\max(z_1, 2 * z_2))$
Gdzie:

d_{max} - maksymalna odległość między robotami

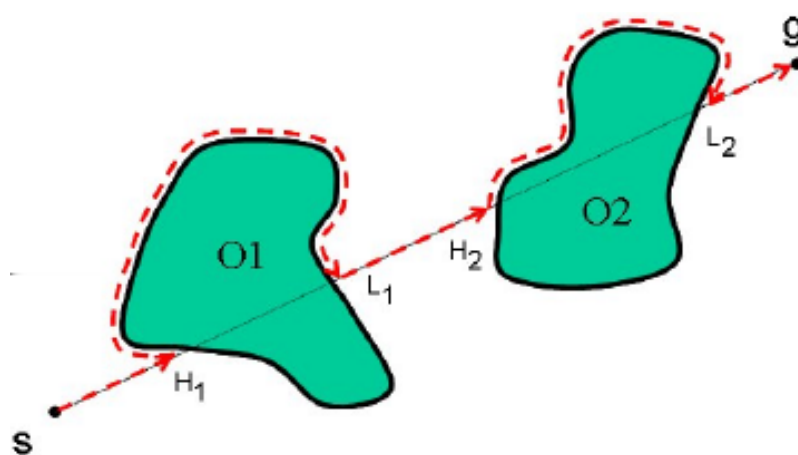
z_1 - zasięg urządzeń komunikacyjnych robotów

z_2 - zasięg czujników odległości

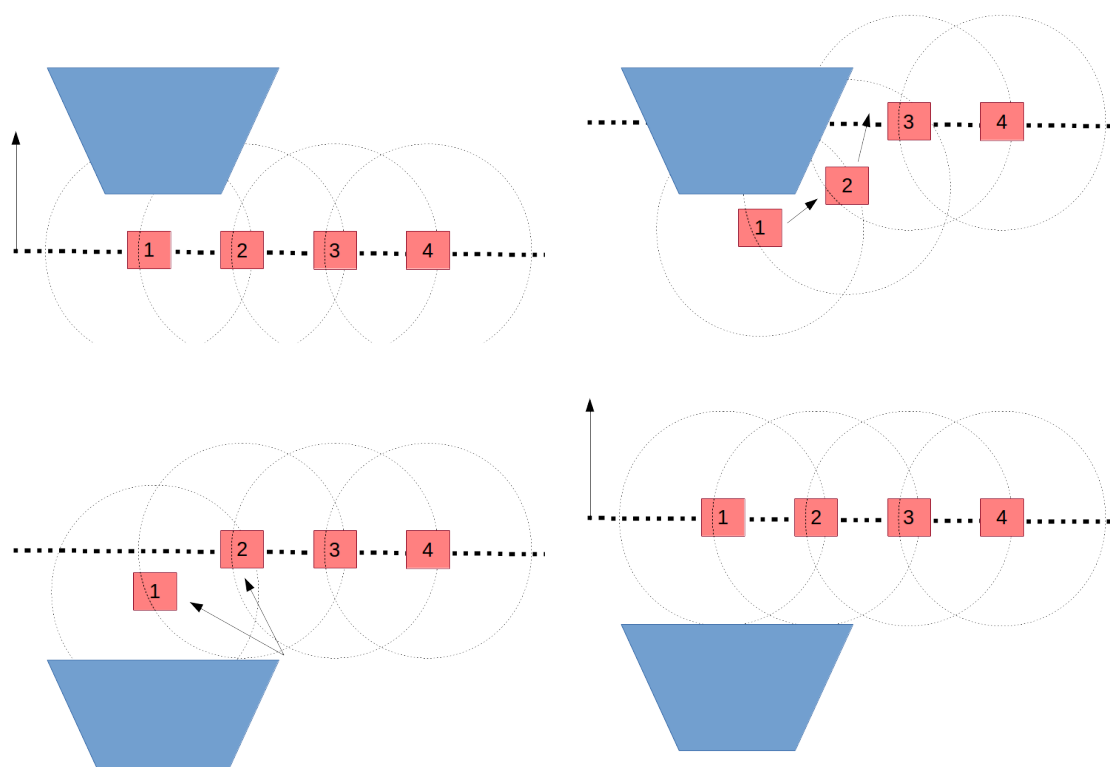
Roboty w ten sposób zaczynają poruszać się do przodu do momentu dotarcia do przeszkody, toksycznej chmury lub granicy obszaru poszukiwań. W przypadku powyższych wydarzeń, uruchamiane są odpowiednie algorytmy. Gdy roboty dotrą do końca obszaru poszukiwań, zatrzymują się, przesuwają w bok i kontynuują poszukiwania w drugą stronę, granicząc z poprzednim obszarem poszukiwań.

3.3.2 Algorytm bug 2

Algorytm ten będzie stosowany w przypadku napotkania przeszkody przez robota. Polega on na podążaniu wzdłuż ściany przeszkody do momentu osiągnięcia punktu leżącego za napotkaną ścianą, na przedłużeniu początkowej ścieżki ruchu. Jego działanie zostało przedstawione na rysunku 2. W zastosowaniu dla grupy robotów zmodyfikujemy ten algorytm w następujący sposób: Gdy robot/roboty wykryją przeszkodę, oceniane jest jej położenie względem środka szeregu. Większość szeregu powinna poruszać się dalej, natomiast pozostałe roboty powinny zacząć omijać przeszkodę w kierunku większości za pomocą algorytmu bug 2, nie przerywając połączenia z siecią. Jego działanie zaprezentowano na rysunku 3.



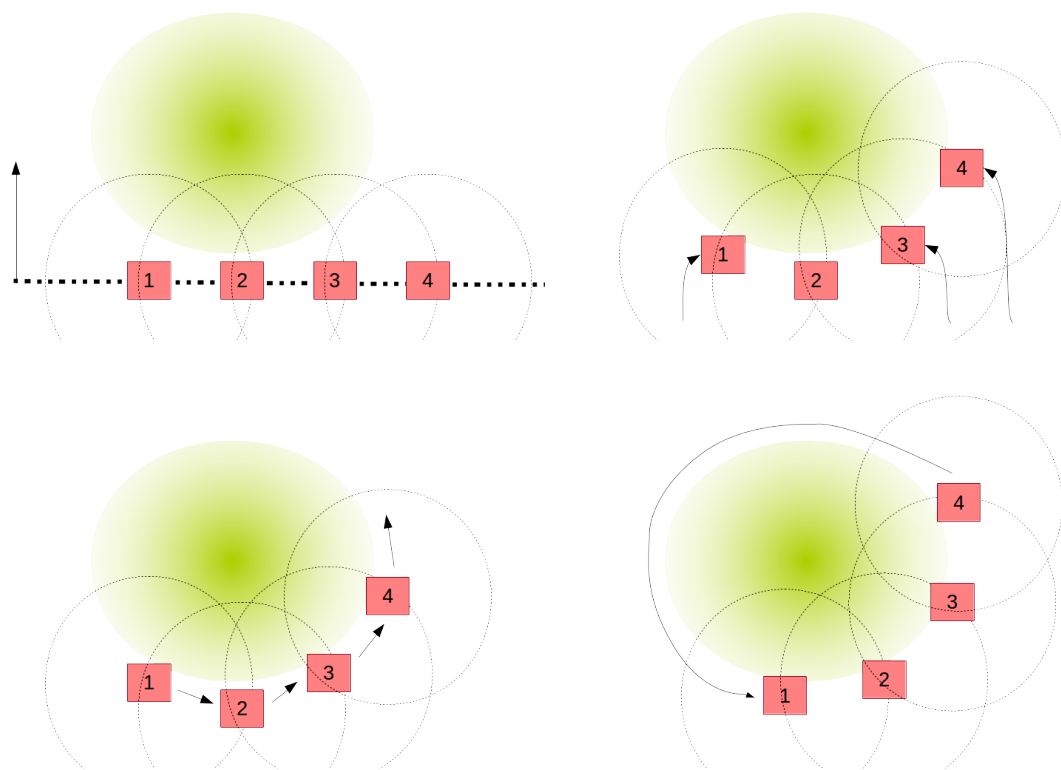
Rysunek 2: Działanie algorytmu bug 2



Rysunek 3: Działanie algorytmu bug 2 dla wielu robotów. Ścianę napotyka jako pierwszy robot 1, robot 2 także ją zauważa. Znajduje się na lewo od środka ciężkości linii, więc roboty z tej strony zaczynają omijać przeszkodę, korzystając z algorytmu bug 2 oraz zachowując spójność sieci.

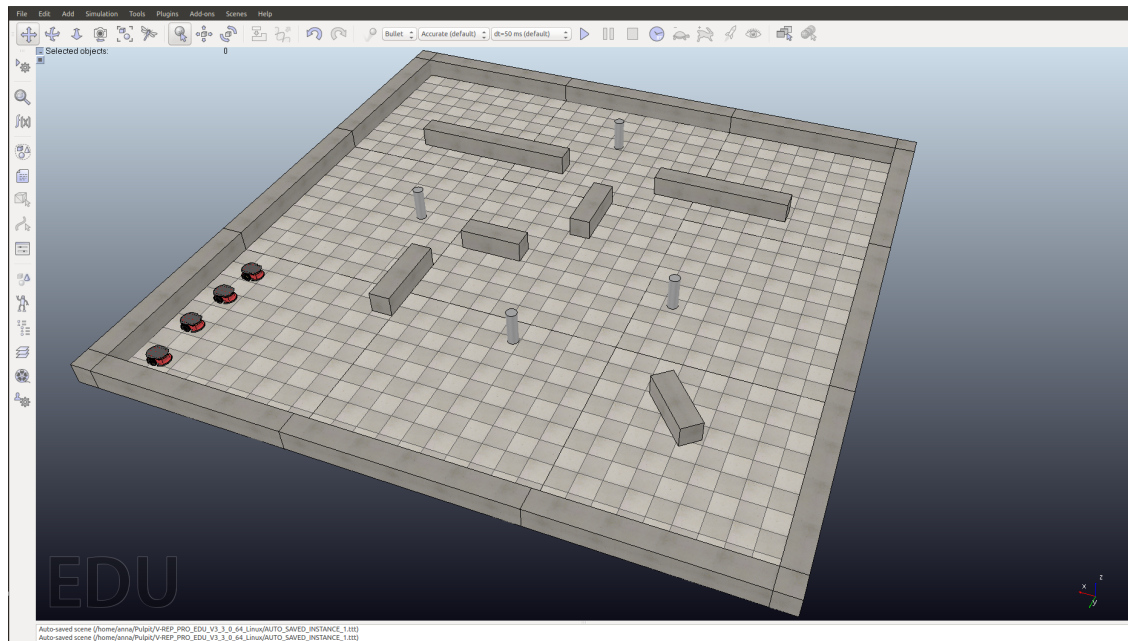
3.3.3 Algorytm otaczania chmury

W momencie napotkania chmury przez któregokolwiek robota, zatrzymuje się on (po osiągnięciu natężenia toksyn określonego programowo). Pozostałe roboty jadą dalej, po łuku – tak, aby nie przerwać połączenia z siecią. Gdy napotkają określone stężenie chmury, zatrzymują się. Gdy wszystkie roboty zatrzymały się, zwracają się w jedną stronę i zaczynają okrążać chmurę, utrzymując się na granicy stężenia oraz zachowując spójność sieci. Jego działanie przedstawiono na rysunku 4.

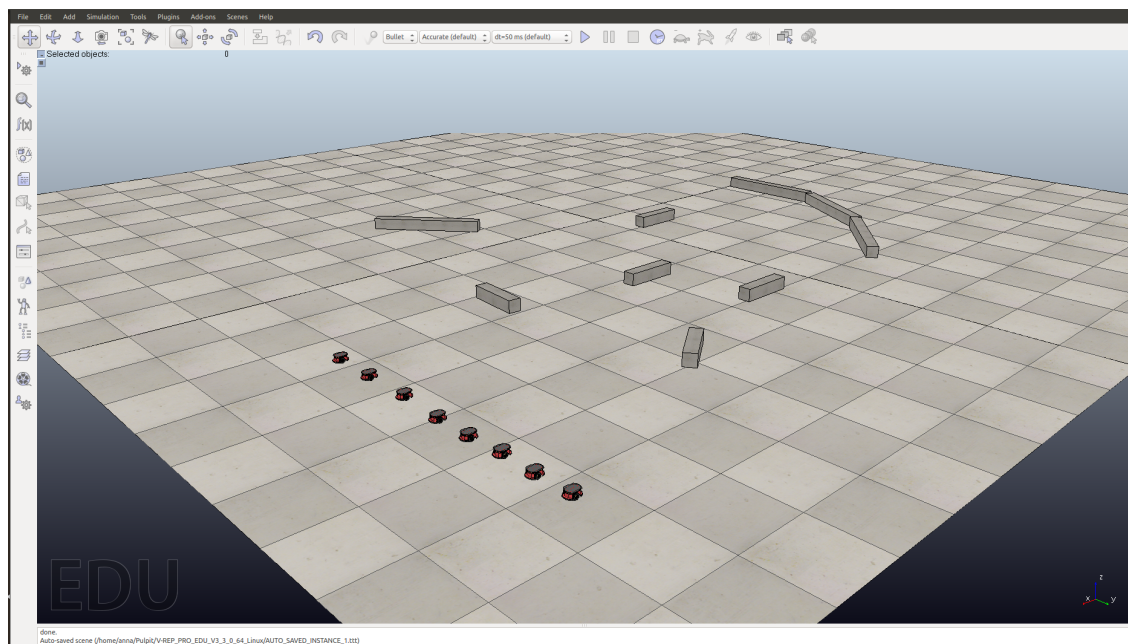


Rysunek 4: Działanie algorytmu otaczania chmury. Robot 2 napotyka chmurę, pozostałe roboty zbliżają się do chmury aż do wartości granicznej, następnie cała sieć zaczyna krążyć wokół chmury, monitorując jej kształt i położenie.

3.4 Przykładowe mapy środowiska



(a) Mały teren



(b) Duży teren

Rysunek 5: Przykładowe mapy środowiska