

# Analiza zagadnienia i przygotowanie środowiska roboczego

Roboty monitorujące skażenie środowiska - symulator V-REP

Anna Wujek  
Łukasz Korpala  
Wiktor Ślęczka

9 maja 2016

# Spis treści

<b>1</b>	<b>Zadanie</b>	<b>3</b>
1.1	Założenia . . . . .	3
1.2	Scenariusz działania . . . . .	3
1.3	Narzędzia . . . . .	3
1.4	Wstępny projekt systemu . . . . .	5
<b>2</b>	<b>Realizacja</b>	<b>5</b>
2.1	Roboty . . . . .	5
2.2	Algorytmy . . . . .	6
2.2.1	Algorytm poszukiwania chmury . . . . .	6
2.2.2	Algorytm bug 2 . . . . .	6
2.2.3	Algorytm otaczania chmury . . . . .	8
2.3	Przykładowe mapy środowiska . . . . .	9

# 1 Zadanie

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

## 1.1 Założenia

- Roboty potrafią się na bieżąco lokalizować.
- System składa się z pewnej liczby robotów mobilnych oraz jednostki centralnej odbierającej od robotów dane o skażeniu.
- 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, z pewnymi ograniczeniami.
- Chmura skażenia jest spójna, może się przemieszczać i zmieniać kształt.

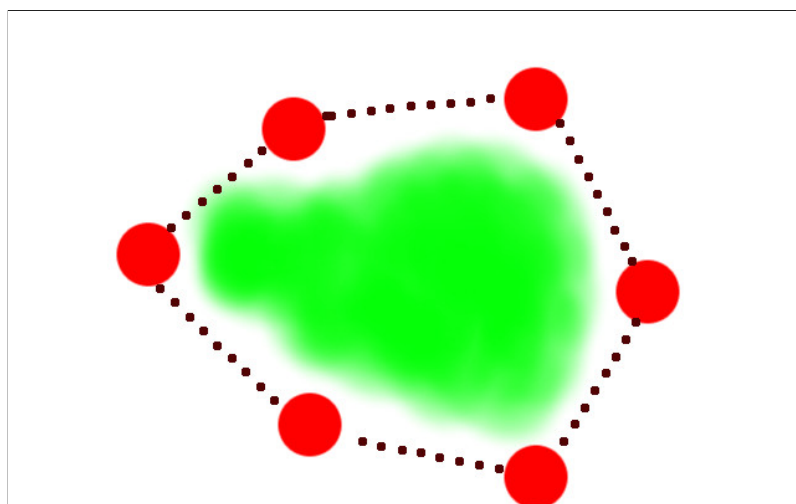
## 1.2 Scenariusz działania

Nieznana jest mapa przestrzeni roboczej oraz lokalizacja chmury skażenia. Roboty są autonomiczne w zakresie omijania przeszkód, a także same organizują się w sieć, a do jednostki centralnej tylko wysyłają dane o skażeniu. Ważne jest w tym przypadku dbanie o zachowanie spójności sieci, zwłaszcza podczas omijania przeszkód. Chmura może się przemieszczać oraz zmieniać kształt w trakcie zadania. Roboty poszukują granic chmury oraz starają się ją całkowicie otoczyć.

## 1.3 Narzędzia

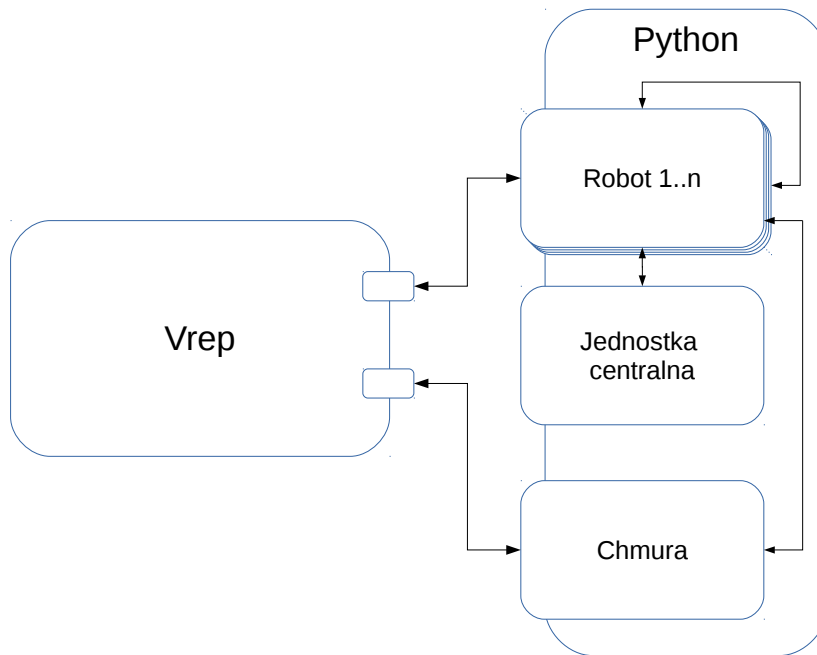
Wykorzystany zostanie symulator V-rep w połączeniu z językiem skryptowym python. V-rep posłuży jako symulator przestrzeni roboczej oraz robotów i ich czujników, natomiast w języku Python zaimplementowana zostanie jednostka centralna oraz logika robotów.

W ramach przygotowania środowiska stworzyliśmy niewielki program do sterowania robotem z zewnętrznego skryptu, który uruchamia silnik robota i odczytuje stan czujników.



Rysunek 1: Przykładowy sposób otoczenia chmury skażenia przez roboty.

## 1.4 Wstępny projekt systemu



Rysunek 2: 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.

## 2 Realizacja

### 2.1 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.

## 2.2 Algorytmy

### 2.2.1 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

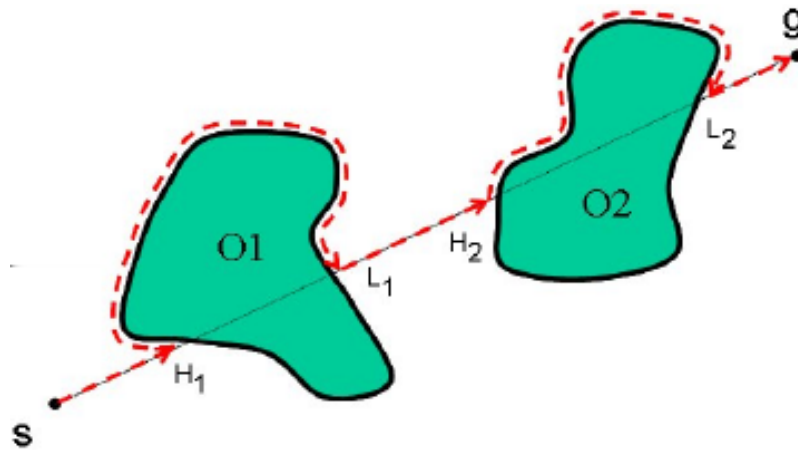
$z_1$  – zasięg urządzeń komunikacyjnych robotów

$z_2$  – 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, przesuwiają w bok i kontynuują poszukiwania w drugą stronę, granicząc z poprzednim obszarem poszukiwań.

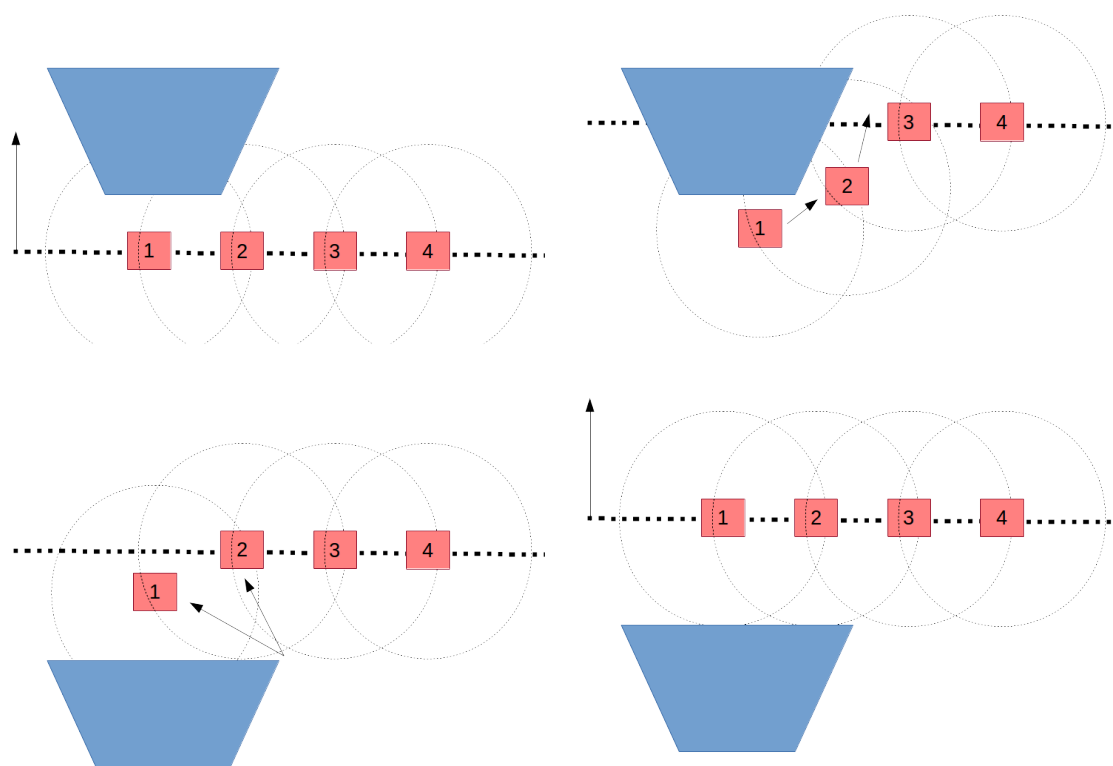
### 2.2.2 Algorytm bug 2

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 3. W zastosowaniu dla grupy robo-



Rysunek 3: Działanie algorytmu bug 2

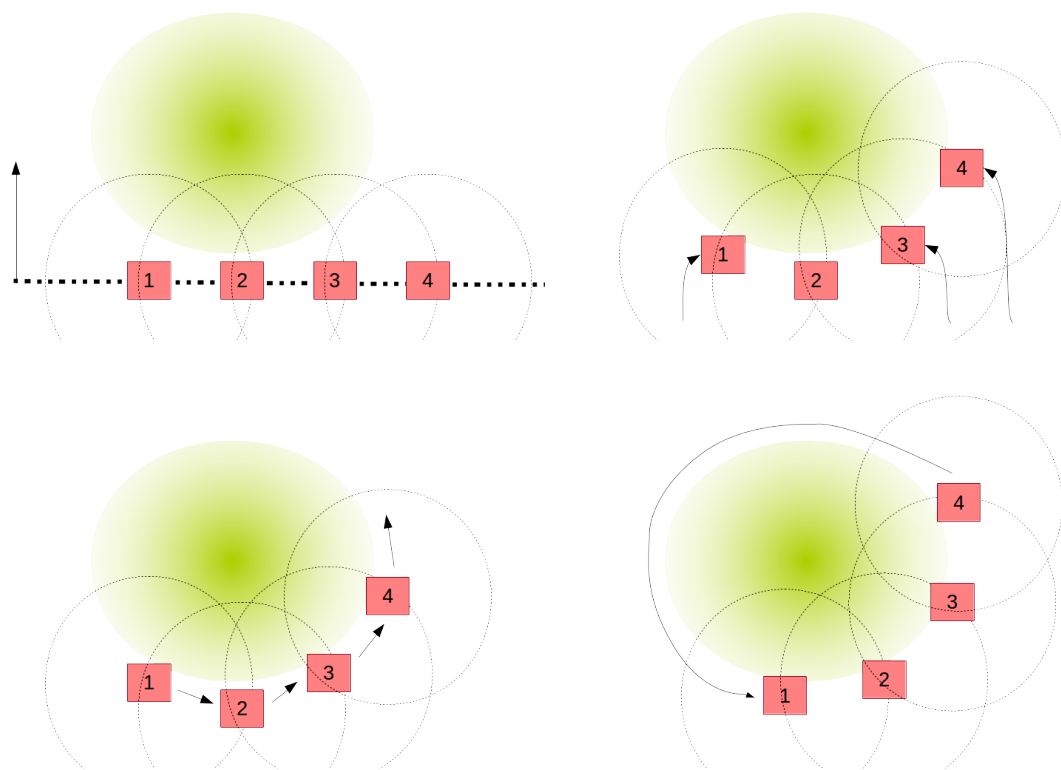
tów zmodyfikujemy ten algorytm w następujący sposób: Gdy robot/roboty wykryją przeszkodę, oceniane jest jej położenie względem środka szeregu. Większość szeregu powinna poruszać się dalej, natomiast pozostałe roboty powinny zacząć omijać przeszkodę w kierunku większości za pomocą algorytmu bug 2, nie przerywając połączenia z siecią. Jego działanie zaprezentowano na rysunku 4.



Rysunek 4: Działanie algorytmu bug 2 dla wielu robotów

### 2.2.3 Algorytm otaczania chmury

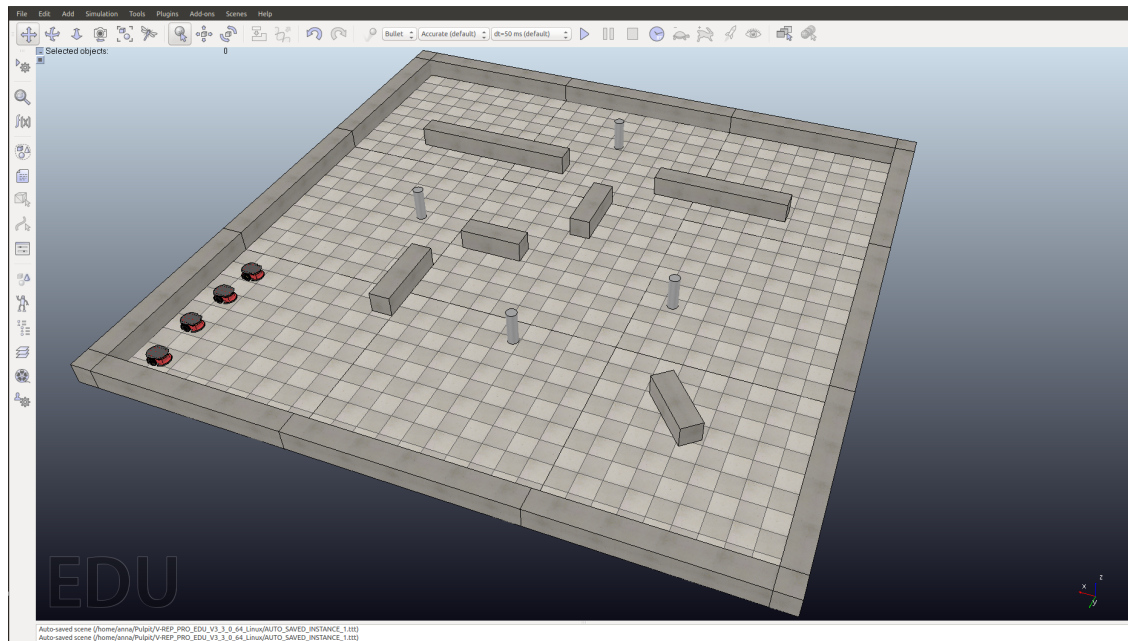
W momencie napotkania chmury przez któregokolwiek robota, zatrzymuje się on (po osiągnięciu natężenia toksyn określonego programowo). Pozostałe roboty jadą dalej, po łuku – tak, aby nie przerwać połączenia z siecią. Gdy napotkają określone stężenie chmury, zatrzymują się. Gdy wszystkie roboty zatrzymały się, zwracają się w jedną stronę i zaczynają okrążyć chmurę, utrzymując się na granicy stężenia oraz zachowując spójność sieci. Jego działanie przedstawiono na rysunku 5.



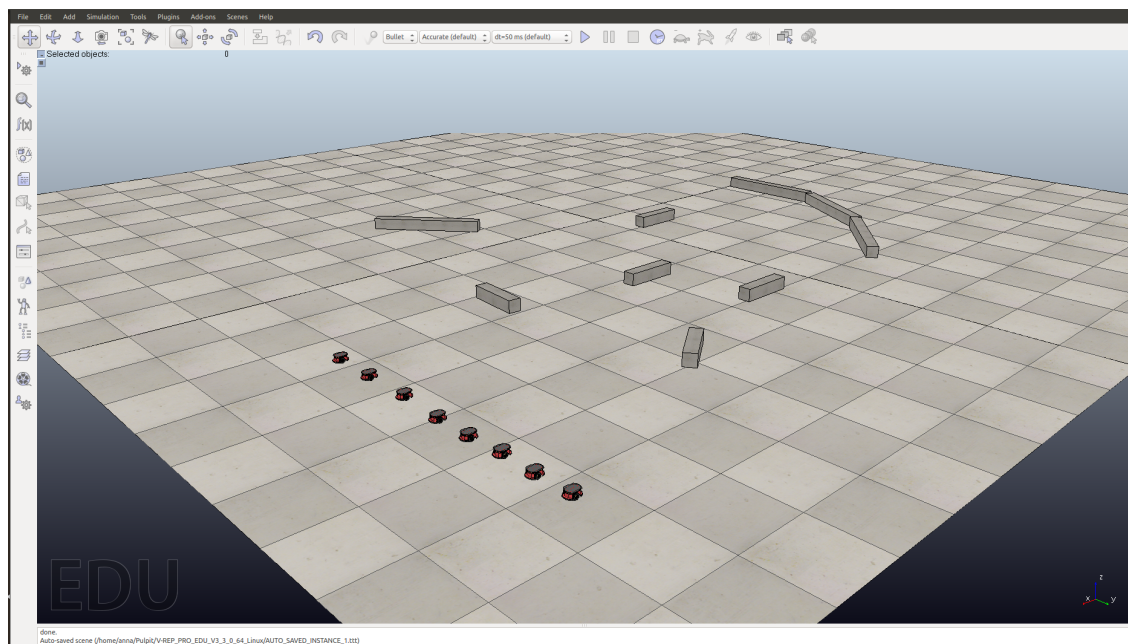
Rysunek 5: Działanie algorytmu otaczania chmury



## 2.3 Przykładowe mapy środowiska



(a) Mały teren



(b) Duży teren

Rysunek 6: Przykładowe mapy środowiska