

POLITECHNIKA WARSZAWSKA

WYDZIAŁ ELEKTRONIKI I TECHNIK INFORMACYJNYCH

Programowanie Równoległe I Rozproszone

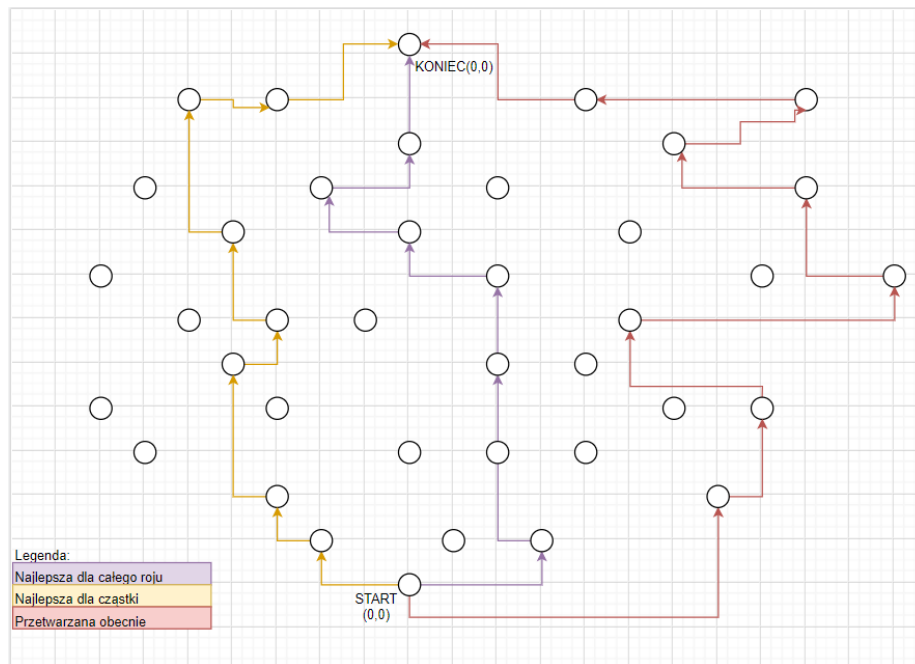
**Metoda optymalizacji rojem cząstek (PSO) do wyznaczenia optymalnej
ścieżki dojazdu do celu dla robota mobilnego.**

Adam Barej, Tomasz Zieliński, Jakub Banaszkiewicz

1. Opis optymalizowanego algorytmu

Zadanie postawione w projekcie polega na wyznaczeniu najkrótszej (w sensie odległości) ścieżki dojazdu do wskazanego punktu w przestrzeni roboczej przez robota mobilnego. Robot porusza się po przestrzeni bazującej na grafie. Do celu prowadzi kilka konkurencyjnych ścieżek.

W celu wykorzystania potencjału algorytmu roju cząstek *Particle Swarm optimization* zdecydowaliśmy się na scenariusz, gdzie robot szuka optymalnej ścieżki, wpuszczając do grafu rój cząsteczek. Cząsteczki nawzajem się ze sobą komunikując, ustalają optymalną ścieżkę przejazdu dla robota. Bazują one na próbie minimalizacji kosztu dotarcia do narzuconego z góry końca trasy.



Rysunek 1 Wizualizacja działania algorytmu

Każdy utworzony graf ma węzeł startowy w punkcie (0,0) i końcowy w punkcie (1, 1). Przy jego tworzeniu wymagane jest podanie przez użytkownika liczby węzłów, które są losowane.

Użytkownik podaje również liczbę cząsteczek roju, biorących udział w procesie doboru najlepszej ścieżki dojazdu z punktu startowego do punktu końcowego na grafie przez robota mobilnego.

Każda z cząsteczek otrzymuje losowy zestaw ścieżek. Zostaje wyznaczona najlepsza obecnie ścieżka dla cząstki i najlepsza obecnie ścieżka dla całego roju.

Jakość ścieżek liczy się jako suma odległości między kolejnymi węzłami. Graf i ścieżki łączące jego węzły znajdują się w przestrzeni 2-wymiarowej.

Podczas kolejnych iteracji działania algorytmu, kolejne węzły przetwarzanej w tej chwili ścieżki są przesuwane w stronę ścieżki optymalnej globalnie i dla cząstki (kolejne węzły są podmieniane na lepsze sąsiednie).

Jeżeli w zestawie ścieżek cząstki pojawi się ścieżka o mniejszym koszcie niż jedna z najlepszych, staje się ona najlepszą (dla cząstki lub globalnie).

Wynikiem działania algorytmu jest koszt najlepszej wyznaczonej ścieżki.

Algorytm działa przez narzuconą z góry liczbę iteracji.

2. Wykorzystana technologia

Algorytm został opracowany w języku programowania C++.

Zbieranie danych z algorytmu i przetwarzanie danych zostało napisane z wykorzystaniem języka Python w wersji 3.6.

Projekt został przygotowany z wykorzystaniem systemu operacyjnego linux ubuntu 18.0.

Do realizacji zadania zrównoleglania z pamięcią wspólną wykorzystano wieloplatformowy interfejs programowania OpenMP.

3. Wyniki

a. Wersja sekwencyjna na pojedynczym procesorze

W celu sprawdzenia powtarzalności danych, przeprowadzono 50 symulacji z następującymi parametrami:

Cecha	Wartość
Graf	Graph2
Liczba iteracji	100
Liczba wątków	1

Otrzymano następujące wyniki działania algorytmu:

Koszt wybranej ścieżki			Czas wykonania algorytmu		
Min	Mean	Max	Min	Mean	Max
1.43341	1.453108	1.48589	1024266.0	1327608.1	2066622.0

Przy jednej iteracji koszt ścieżki oscyluje w granicach 1.52

Średnia wartość wyniku wyniosła:

b. Wersja równoległa z pamięcią wspólną

W programie zrównoleglono 3 fragmenty algorytmu:

Przeprowadzono symulację z wykorzystaniem zrównoleglania z pamięcią wspólną.

W celu oceny jakości procesu zrównoleglania zostanie obliczony współczynnik przyspieszenia oraz wydajność:

Współczynnik przyspieszenia (ang. speed-up) zadania o wielkości n – dzięki zrównolegleniu na p procesorów

$$S(n, p) = T(n, 1) / T(n, p)$$

Konieczne jest spełnienie nierówności

$$S(n, p) \leq p$$

Wydajność to współczynnik przyspieszenia podawany w wersji przeskalowanej w stosunku do przyspieszenia idealnego

$$S(n, p) = S(n, p) p \cdot 100\%$$

n - parametr opisujący wielkość zadania. W naszym przypadku ustalony na 1 przy tych samych warunkach wejściowych pomiędzy testami,

p – liczba procesorów,

$T(n, p)$ – czas wykonania programu realizującego ten sam algorytm dla zadania o wielkości n na maszynie z p procesorami,

- wygenerowanie cząstek z losowym zestawem ścieżek
- ewolucję ścieżek poszczególnych cząstek w celu znalezienia optymalnego rozwiązania
- szukanie węzłów zbliżających cząstkę do optymalnego rozwiązania