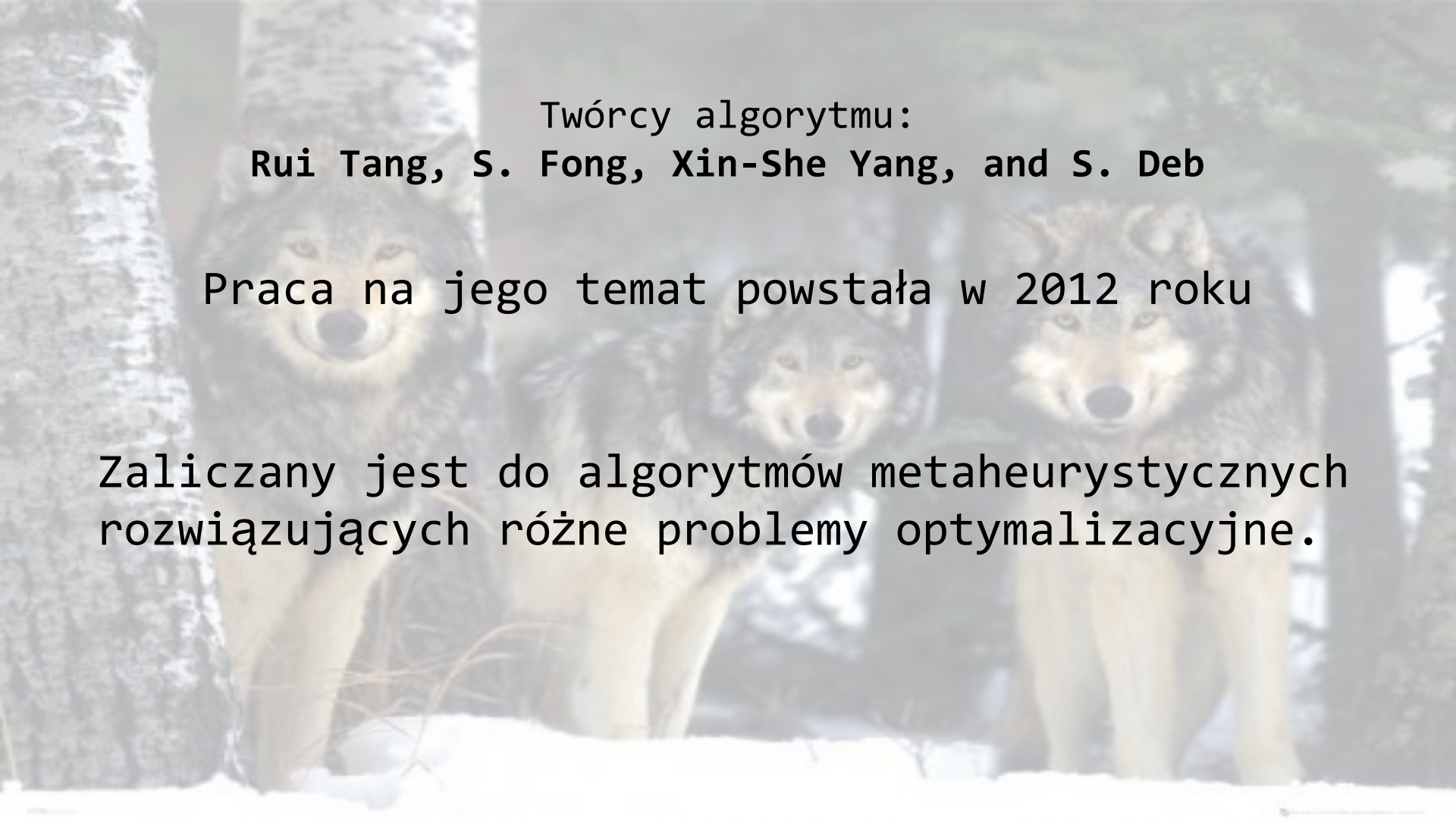




Wolf Search Algorithm

Igor Szylak
Alona Skyba
21 kwietnia 2020

A photograph of three wolves standing in a snowy forest. The wolves are positioned in the background, looking towards the camera. The scene is dimly lit, with snow covering the ground and trees. The text is overlaid on this image.

Twórcy algorytmu:
Rui Tang, S. Fong, Xin-She Yang, and S. Deb

Praca na jego temat powstała w 2012 roku

Zaliczany jest do algorytmów metaheurystycznych rozwiązujących różne problemy optymalizacyjne.

Inspiracja

- Algorytm ten jest inspirowany zachowaniem wilków poszukujących jedzenia i unikających wrogów.



Wstęp teoretyczny

Wilki są drapieżnikami socjalnymi które polują w watahach. W odróżnieniu od algorytmów takich jak PSO czy Fish Swarm w których agenci poruszają się w relatywnie dużych skupiskach, wilki przemieszczają się zazwyczaj jako “rodziny nuklearne” tj. składające się przeważnie z dwóch pokoleń - rodziców i ich dzieci. Zwierzęta te opierają polowania głównie na doskonałym zmyśle węchu który, w algorytmie, jest reprezentowany jako zasięg wykrywania ofiary i innych wilków swojej watahy.

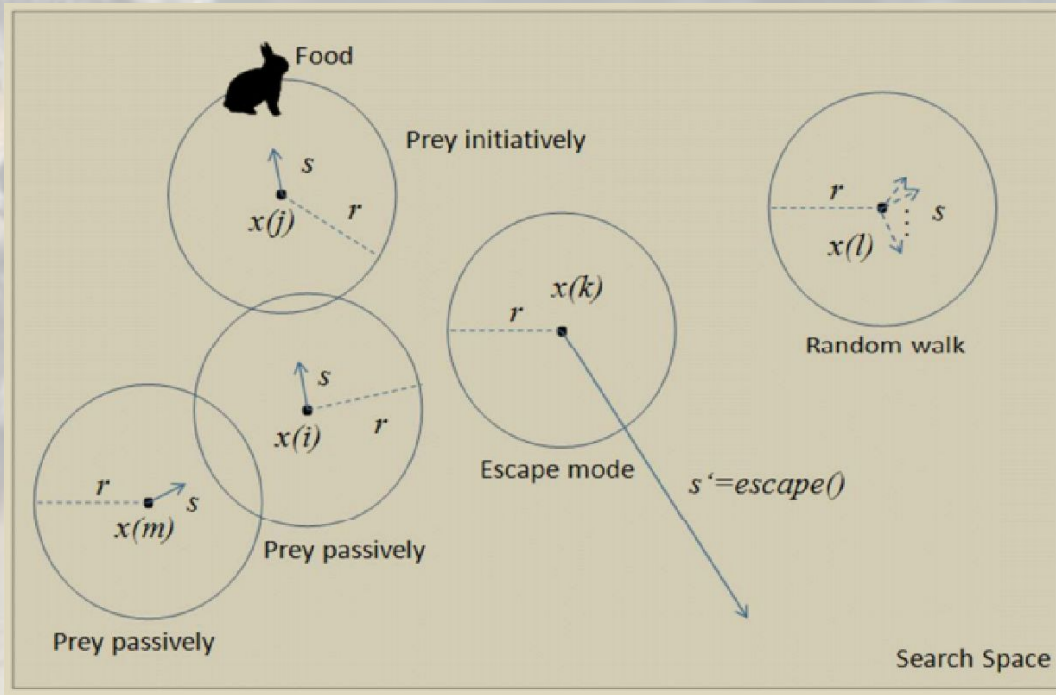
Podstawowe zasady algorytmu

- Każdy wilk posiada stały promień wykrywania definiowany przez \mathbf{v} dla zbioru \mathbf{X} rozwiązań.
- Funkcja celu $f(\mathbf{x})$ dla $\mathbf{x} = (x_1, x_2, \dots, x_d)^T$ określa jakość aktualnej pozycji wilka. Wilk zawsze stara się poprawić swoją pozycję, jednak zamiast wybierać najlepszą możliwą, szuka pozycji która jest lepsza od obecnej i w której znajduje się już wilk z jego watahy.
- Wilk wyczuwszy wroga ucieknie do nowej losowej pozycji z dala od zagrożenia

Pseudokod

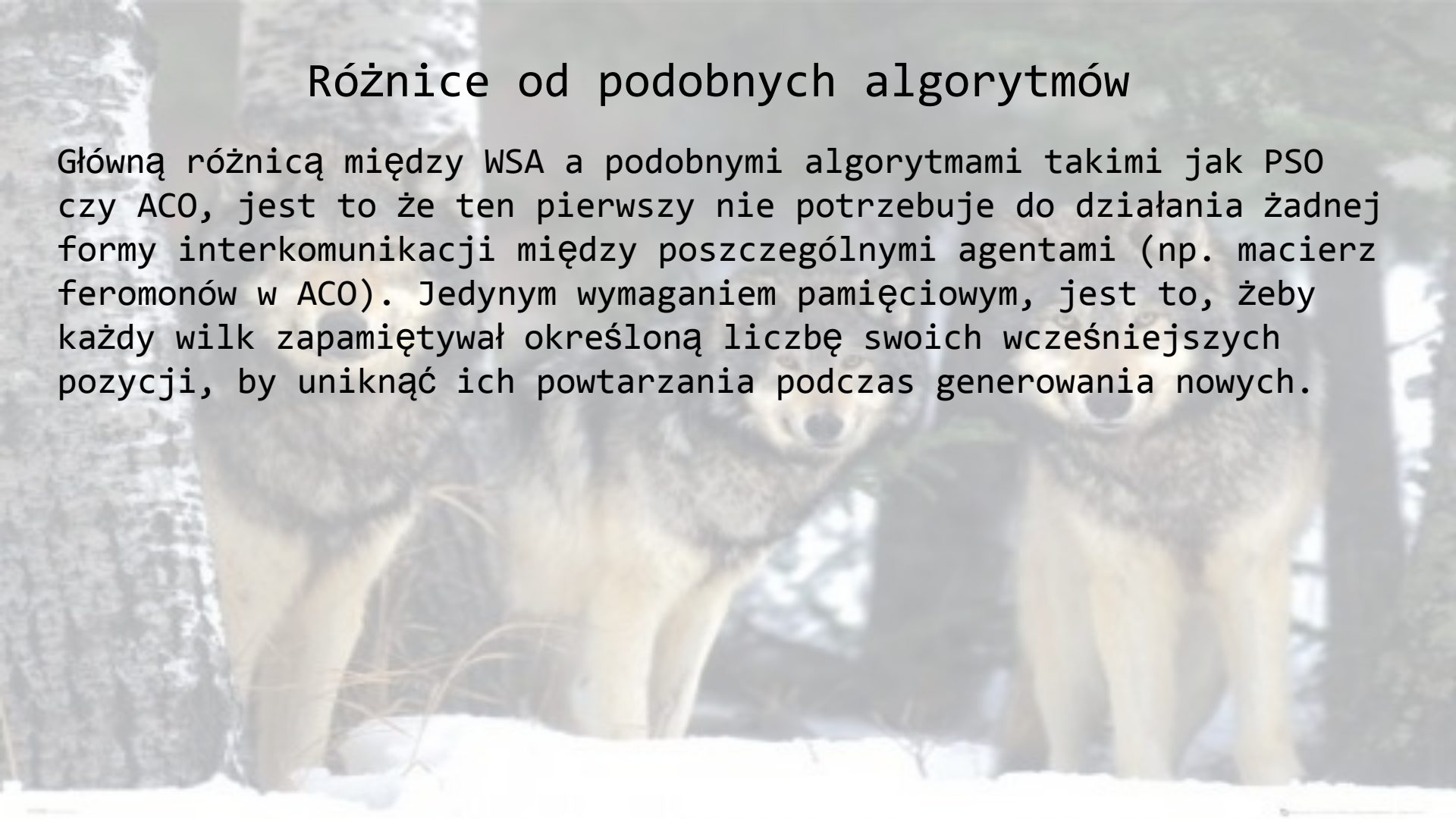
```
Objective function  $f(x)$ ,  $x=(x_1, x_2, \dots, x_d)^T$   
Initialize the population of wolves,  $x_i (i=1, 2, \dots, W)$   
Define and initialize parameters:  
 $r$  = radius of the visual range  
 $s$  = step size by which a wolf moves at a time  
 $\alpha$  = velocity factor of wolf  
 $p_a$  = a user-defined threshold [0..1], determines how  
frequently an enemy appears  
  
WHILE ( $t < \text{generations}$  && stopping criteria not met)  
  FOR  $i=1:W$  // for each wolf  
    Prey_new_food_initiatively();  
    Generate_new_location();  
    // check whether the next location suggested by  
the random number generator is new. If not, repeat  
generating random location.  
    IF ( $\text{dist}(x_i, x_j) < r$  &&  $x_j$  is better as  $f(x_i) < f(x_j)$ )  
       $x_i$  moves towards  $x_j$  //  $x_j$  is a better than  $x_i$   
    ELSE IF  
       $x_i = \text{Prey\_new\_food\_passively}()$ ;  
    END IF  
    Generate_new_location();  
    IF ( $\text{rand}() > p_a$ )  
       $x_i = x_i + \text{rand}() + v$ ; // escape to a new pos.  
    END IF  
  END FOR  
END WHILE
```

Schemat graficzny



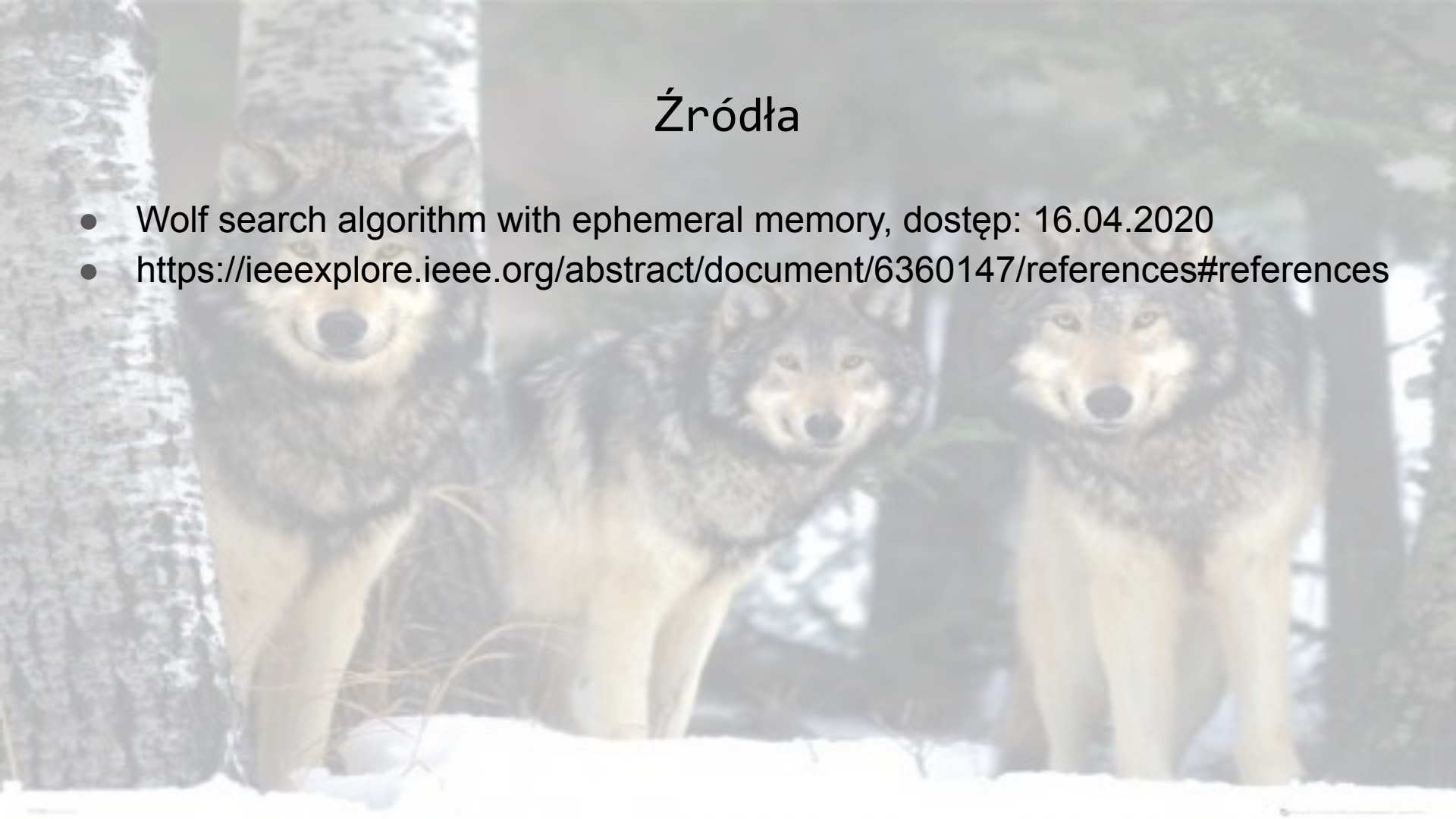
Różnice od podobnych algorytmów

Główną różnicą między WSA a podobnymi algorytmami takimi jak PSO czy ACO, jest to że ten pierwszy nie potrzebuje do działania żadnej formy interkomunikacji między poszczególnymi agentami (np. macierz feromonów w ACO). Jedynym wymaganiem pamięciowym, jest to, żeby każdy wilk zapamiętywał określoną liczbę swoich wcześniejszych pozycji, by uniknąć ich powtarzania podczas generowania nowych.



Źródła

- Wolf search algorithm with ephemeral memory, dostęp: 16.04.2020
- <https://ieeexplore.ieee.org/abstract/document/6360147/references#references>



A photograph of three wolves standing in a snowy forest. The wolves are positioned in a line, with the leftmost one partially obscured by a tree trunk. They are looking towards the camera. The background is a dense forest of evergreen trees covered in snow. The text "Dziękujemy za uwagę!" is overlaid in the center of the image.

Dziękujemy za uwagę!