Mrówcza robota

Marcin Mazur

Obserwacje przyrodnicze pokazują, że jeżeli z mrowiska do źródła pożywienia prowadzą dwie drogi – jedna dłuższa, druga krótsza – to po pewnym czasie większość mrówek będzie korzystać z tej krótszej. Jest to konsekwencją tego, iż pomimo początkowo losowego charakteru ich wędrówki, po znalezieniu pożywienia wracają one do mrowiska pozostawiając na swojej trasie tzw. *ślad feromonowy*. Okazuje się, że na krótszej trasie siła działania feromonów jest większa, więc mrówki wybierają ją chętniej, przez co atrakcyjna ścieżka staje się coraz bardziej atrakcyjna, a mniej atrakcyjna z czasem traci na znaczeniu. Podsumowując, zachowanie mrówek można opisać następującym algorytmem:

- mrówki są głodne, więc szukają pożywienia;
- na początku wędrują w sposób losowy;
- po znalezieniu pożywienia wracają do mrowiska pozostawiając na swojej trasie ślad feromonowy;
- feromony stopniowo parują;
- na krótszej trasie siła działania feromonów jest większa, więc mrówki wybierają ją chętniej, wzmacniając ślad feromonowy;
- atrakcyjna ścieżka staje się coraz bardziej atrakcyjna, a mniej atrakcyjna z czasem traci na znaczeniu.

Czy wzorując się na takim zachowaniu mrówek można stworzyć ogólną technikę algorytmiczną, służącą do rozwiązywania trudnych problemów optymalizacyjnych (jakich)?

Odpowiedzią na postawione powyżej pytanie jest metoda zwana *optymalizacją kolonii mrówek (ang. Ant Colony Optimization, ACO)*. Pierwszy system ACO został zaproponowany w 1992 roku przez Marco Dorigo. Od tego czasu technika ta podlegała (i wciąż podlega) licznym modyfikacjom i ulepszeniom. W dalszej części postaram się pokazać zasadę działania algorytmu mrówkowego w rozwiązaniu tzw. *problemu komiwojażera (ang. Travelling Salesman Problem).*

Oględnie mówiąc, problemem komiwojażera (wędrownego sprzedawcy) jest znalezienie najkrótszej (najtańszej, najszybszej) drogi łączącej miasta, które zamierza on odwiedzić. Ujmując rzecz matematycznie, chodzi o znalezienie minimalnego (w sensie sumy wag krawędzi) cyklu Hamiltona (czyli zamkniętej ścieżki przechodzącej przez wszystkie wierzchołki grafu) w pełnym grafie ważonym (tzn. takim, w którym każde dwa wierzchołki są połączone krawędzią oraz każdej krawędzi przypisana jest waga). Algorytm mrówkowy dla problemu komiwojażera można (w uproszczeniu) zapisać w następujący sposób:

BEGIN

zainicjuj początkowy (niewielki) poziom feromonu na krawędziach grafu;

REPEAT

odparuj część feromonu ze wszystkich krawędzi;

ustaw mrówki w losowych wierzchołkach;

wygeneruj ścieżki dla wszystkich mrówek na podstawie aktualnego poziomu feromonu na krawędziach grafu: kolejne wierzchołki na trasie każdej mrówki wybierane są losowo (bez powtórzeń!), jednakże preferowane są przejścia po krawędziach krótszych oraz tych z większą ilością feromonu;

zapamiętaj (jako rozwiązanie) najkrótszą znalezioną dotychczas ścieżkę;

zwiększ poziom feromonu na krawędziach grafu, po których przeszły mrówki, o wartość odwrotnie proporcjonalną do całkowitej długości trasy danej mrówki;

UNTIL

zachodzi warunek kończący (stabilizacja rozwiązania, przekroczenie maksymalnej liczba iteracji);

END

Na zakończenie warto podkreślić, iż szerokim polem zastosowań algorytmu mrówkowego są wszelkiego rodzaju zagadnienia optymalizacyjne związane z transportem.