KÜNSTLICHE INTELLIGENZ

LABOR 6

AUFGABE 1

- Schwarmsystene sind asynchrone Systeme. Es gibt keine zentrale Uhr, auf die jeder zugreifen könnte. Falls ein Schwarm dennoch synchron agieren muss, dann muss er sich zuerst explizit synchronisieren. Ein Beispiel eines biologischen Systems ist eine Population von Glühwürmchen. Bei manchen Spezies ist bekannt, dass sie sich selbstsynchronisieren

Im Zusammenhang dazu steht die Forschung aus den Bereichen der drahtlosen Sensornetze und Adhoc-Netzwerke und deren dezentrale Synchronisierung

AUFGABE 1

- Im Folgenden bauen wir uns ein einfaches Modell einer solchen Glühwürmchenpopulation. Die Population ist über ein Einheitsquadrat (1×1) zufällig verteilt (Gleichverteilung).Wir nehmen an, dass die Glühwürmchen stationär sind und nur die Nachbarn in ihrer Nähe wahrnehmen können. Wir sagen, dass zwei Glühwürmchen sich gegenseitig sehen können, wenn die Distanz zwischen den beiden kleiner als r ist.

AUFGABE 1

- Wir haben wir eine virtuelle Scheibe mit Radius *r* , deren Mitte an der Position des Glühwürmchens liegt, und jedes andere Glühwürmchen, das auf der Scheibe sitzt, ist ein Nachbar. Die Glühwürmchen leuchten in Zyklen auf. Wir definieren die Länge des Zyklus als $L = 50$ Zeitschritte. Ein Glühwürmchen leuchtet für $L/2$ Zeitschritte auf, gefolgt von $L/2$ Zeitschritten ohne Leuchten. Das gilt immer außer für den Fall, dass das Glühwürmchen versucht seinen Zyklus zu synchronisieren. Wenn es angefangen hat zu leuchten, überprüft es im folgenden Zeitschritt seine Nachbarn und testet, ob die Mehrheit der Nachbarn bereits leuchtet. Wenn dies der Fall ist, dann korrigiert das Glühwürmchen seine interne Uhr, indem es eins hinzuzählt. Damit reduziert es die Zeit seines aktuellen Leuchtzyklus von $L/2$ auf $L/2 - 1$ Zeitschritte und wird daher in der nächsten Runde einen Zeitschritt früher aufleuchten.

AUFGABE 1, TEIL A

- Implementiere das Modell für eine Schwarmgröße von $N = 150$ und eine Zykluslänge von $L = 50$. Berechne die durchschnittliche Anzahl von Nachbarn pro Glühwürmchen für die Nachbarschaftsradien $r \in \{0{,}05;\ 0{,}1;\ 0{,}5;\ 1{,}4\}$. Plotte die Anzahl von momentan leuchtenden Glühwürmchen über die Zeit für verschiedene Radien $r \in \{0{,}05;\ 0{,}1;\ 0{,}5;\ 1{,}4\}$ für jeweils 5000 Zeitschritte. Achte beim Plotten der momentan leuchtenden Glühwürmchen darauf, das volle Intervall [0; 150] für die vertikale Achse anzuzeigen.

AUFGABE 1, TEIL B

Erweitere Dein Modell, um Minimum und Maximum der momentan leuchtenden Glühwürmchen während des allerletzten Zyklus (also die letzten $L = 50$ Zeitschritte ab $t = 4950$) zu bestimmen. Wenn wir das Minimum vom Maximum abziehen, bekommen wir die doppelte Amplitude der Synchronisationswelle. Ermittle den Durchschnitt der gemessenen Amplituden für jeweils 50 Stichproben (50 unabhängige Simulationsläufe mit jeweils 5000 Zeitschritten) und plotte diese für verschiedene Radien $r \in [0{,}025;\, 1{,}4]$ in Schritten der Größe 0,025. Was ist eine gute Wahl für den Nachbarschaftsradius und somit die Schwarmdichte?

ANT COLONY OPTIMIZATION

Implementiere die Ant Colony Optimization (ACO) und versuche eine Instanz des Problems des Handlungsreisenden zu lösen. Dazu kann man entweder selbst Probleminstanzen generieren und zufällig Städte im Raum verteilen oder man kann bekannte Probleminstanzen herunterladen (z.B. von hier: https://wwwproxy.iwr.uni-heidelberg.de/groups/comopt/software/TSPLIB95/tsp/) was den Vorteil hat, dass man dann die beste Route kennt und die Leistung des eigenen Algorithmus abschätzen kann. Experimentiere mit den verschiedenen Parametern und untersuche deren Einfluss auf die Leistung. Plotte die Länge der bisher besten gefundenen Lösung über die Zeitschritte.