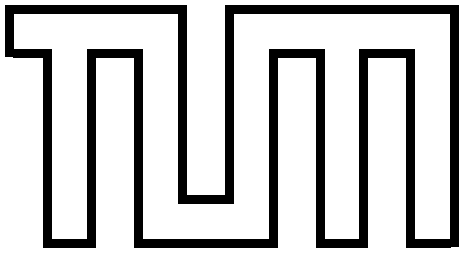

Technische Universität München

Fakultät für Informatik

Forschungs- und Lehreinheit Informatik VI

Selbstorganisierende Karten

Seminar Kognitive Robotik (SS12)

Thomas Hoffmann

Betreuer: Dr. Florian Röhrbein

Leitung: Prof. Alois Knoll

Abgabetermin: 21. Juli 2012

Inhaltsverzeichnis

1 Einleitung

Ziel dieser Seminararbeit ist es einen Überblick über Selbstorganisiernde Karten zu geben. Dazu wird ausgehend von einer allgemeinen Beschreibung Künstlicher Neuronaler Netze, die Selbstorganisierende Karte in diesen Themenkomplex einsortiert. Anschließend wird die Funktionsweise erläutert und einige Anwendungen aufgezeigt.

1.1 Allgemeines zu Neuronalen Netzen und Unüberwachtem Lernen

1.1.1 Neuronale Netze in biologischen Systemen

Das Vorbild für Künstliche Neuronale Netze findet sich in biologischen neuronalen Netzen wie beispielsweise dem menschlichen Gehirn. Zentrales Element in diesen System ist das Neuron.

Bei Neuronen handelt es sich um Körperzellen, welche Signale auf elektrischem und chemischem Weg verarbeiten und weiterleiten. Die Dendriten sind verzweigte Auswüchse der Zelle, welche über eine synaptische Verbindung, die Reize chemisch überträgt, von anderen Körperzellen angeregt werden können. Bei einem bestimmten Schwellenwert entlädt sich das elektrische Potenzial der Zelle über das Axon. Dieses leitet den elektrischen Impuls zu synaptischen Verbindungen, von wo das Signal an andere Zellen weitergegegeben wird. Die Dendriten sind dabei eingehende NervenBahnen die Signale zum Neuron bringen, das Axon leitet Signale an andere Zellen weiter. Körperzellen mit den das Neuron interagiert sind meist selbst Nervenzellen, eskann sich aber auch um Sinneszellen an den Dendriten und Nervenfasern zu Muskeln am Axon handeln. Auf diese Weise kann das Gehirn den Körper steuern.[4]

Die synaptischen Verbindungen zwischen Neuronen variieren hinsichtlich der Stärke der Signalübertragung. Hierbei besteht eine Dynamik die es ermöglicht das Verbindungen durch äußere Einflüsse verstärkt oder abgeschwächt werden. Ebenso ist es möglich das Verbindungen ganz verschwinden und neue geknüpft werden. Auf diese Weise kann sind neuronale Netze adaptiv und können daher Lernvorgänge durchführen.[3]

1.1.2 Künstliche Neuronale Netze(KNN)

Aufgrund der Vorteile die sich durch die hohe Parallelität neuronaler Netze ergeben, wurde auch in der KI-Forschung begonnen Ansätze zu verfolgen welcher auf Übertragung der Lernsysteme in biologischen Netzen auf künstliche neuronale Netze (KNNs) beruhen. Diese werden dazu genutzt Problemstellungen wie Mustererkennung und Klassifikation zu

lösen welche im Allgemeinen von biologischen Systemen besser bewältigt werden als von Computern.[3]

Als Eingabeinformationen für das Neuron n bestehen gewichtete Verbindungen zu den Neuronen 1...j, welche die Ausgabewerte $o_1 ... o_j$ welche mit den Gewichten $w_{1n} ... w_{jn}$ mit dem Neuron verbunden sind. Eine Gewichtsfunktion/Propagierungsfunktion berechnet daraus die Netzeingabe net_n des Neurons. Dafüer wird in der Regel die gewichtete Summe verwendet, sodass sich $net_n = sum_j wjn * o_j$ ergibt. Eine Aktivitätsfunktion f_{act} berechnet daraus die neue Aktivität des Neurons a_n, wobei dabei meist die Netzeingabe auf einen Schwellenwert hin überprüft wird. Die Ausgabefunktion f_{out} berechnet aus der Aktivität des Neurons anschließend den Ausgabewert o_n. Die Ausgabefunktion dient dabei häeufig dazu aus der Aktivität einen binären Ausgabewert herauszufiltern.[5]

Ein neuronales Netz als Ganzes besteht aus einer Menge solcher Neuronen, welche durch gewichtete Verbindungen miteinander verbunden sind. Durch die Art der Verknüpfungen werden unterschiedliche Architekturen definiert. Neuronen welche keine eingehenden Verbindungen besitzen modellieren Eingabe-Neuronen und bilden zusammen die Eingabeschicht. Neuronen ohne ausgehende Verbindungen sind Ausgabe-Neuronen und bilden entsprechend die Ausgabeschicht des neuronalen Netzwerks.[5]

1.1.3 Lernverfahren für KNNs

Unter den neuronalen Netzarchitektur gibt es die Gruppe der trainierbaren neuronalen Netze. Bei trainierbaren Netzen findet ein Lernvorgang dadurch statt, dass Trainingsmuster an die Eingabeschicht angelegt werden und eine, von einem Lernverfahren festgelegte, Anpassung der Verbindungsgewichte erfolgt.[5]

Die hierfür verwendeten Lernverfahren lassen sich in die Kategorien überwachtes, bestärkendes und nicht überwachtes Lernen unterteilen:

- Bei überwachtem Lernen liegen sowohl Eingabemuster als auch die hierzu für die Ausgabe gewünschte Sollmuster vor. Das Training des Neuronalen Netzes findet durch die Korrektur des Ausgabemuster hin zum Sollmuster mittels durch Anpassung von Verbindungsgewichten und Schwellenwerten statt.[4]
- Bei bestärkendem Lernen findet ebenfalls ein solches Training statt wobei zur Adaptierung keine konkreten Ausgabewerte verwendet werden, sondern nur die Information, ob das Ausgabemuster korrekt ist.[5]
- Beim unüberwachten Lernen gibt es kein vorgegebenes Ergebnis zu bestehenden Eingabemustern. Bei diesen Lernverfahren soll im Gegensatz zu den beiden anderen kein Verhalten erlernt werden, welches aus Eingabemustern bestimmte Ausgabemuster generiert. Stattdessen ist es das Ziel ein klassifizierendes Verhalten zu erlernen, das beliebige Eingabemuster bestimmten Kategorien zuordnet. Da aufgrund der unbekannten Klasseneinteilung keine Netzausgänge vorgegeben sind, besitzt ein zugehöriges

Neuronales Netz keine Ausgabeschicht.[4]

1.1.4 Wettbewerbslernen

Wettbewerbslernen fasst Varianten unüberwachter Lernverfahren zusammen bei denen im neuronalen Netz auf einer Wettbewerbsschicht die Aktivierung eines Neurons nicht nur von der eigenen Netzeingabe abhängt sondern auch von den Netzeingaben anderer Neuronen. Gemäß dem dabei verwendeten winner takes it all Vorgehen erhält nur ein einziges Neuron eine positive Aktivität, welches über die maximale Netzeingabe verfügt, alle anderen werden inaktiv gesetzt. Die Neuronen stehen also im Wettbewerb um maximale Netzeingabe, welche von den Gewichten der Verbindungen von der Eingabeschicht abhängen.[6] Eine Unterteilung des Wettbewerbslernens lässt sich in hartes und weiches Wettbewerbslernen vornehmen[6]:

- Beim harten Wettbewerbslernen werden nur die Gewichte des Gewinner-Neurons angepasst, sodass dieses eine noch höhere Netzeingabe beim angelegten Eingabemuster aufweist.
- Beim weichen Wettbewerbslernen werden zusätzlich die Gewichte der Neuronen in der Nachbarschaft des Gewinnerneurons so angepasst, dass auch sie beim angelegten Eingabemuster eine höhere Netzeingabe aufweisen. Somit ist es für weiches Wettbewerbslernen erforderlich, dass die Neuronen in einer räumlichen Ordnung zueinander stehen, sodass Nachbarn zu Neuronen bestimmt werden können.

1.2 Allgemeines zu Selbstorganisierenden Karten

1.2.1 Geschichtliches

Selbstorganisiernde Karten wurden 1982 von Teuvo Kohonen entwickelt und werden daher auch Kohonen-Karten genannt.[3] Ebenso gebräuchlich ist die englische Abkürzung SOM (Self-Organizing-Map).

1.2.2 Klassifizierungsprobleme für KNNs

Die Selbstorganisierende Karte wurde entwickelt um die Problemstellung der Bestimmung von Klassen in Datenmengen nur unter Zuhilfenahme von einer begrenzten Zahl an Eingabedaten, den Trainingsdaten zu erlernen. Dabei besteht jeder Datensatz der Traingingsdaten aus einer festgelegten Zahl von Parametern die auf ein Eingabemuster abgebildet und an das neuronale Netz angelegt werden können. Im Trainingvorgang werden nacheinander die Eingabemuster angelegt und das neuronale Netz soll sich dabei so adaptieren, dass es

in der Lage ist jedes Eingabemuster einer Klasse zuzuordnen. Anwendung findet dies beispielweise bei der Clusteranalyse im Bereich des Data-Minings, wobei dort aus ein großen Datenmenge bestimmte Eigenschaften herausgefiltert werden sollen.

1.2.3 Kortikale Karten im menschlichen Gehirn als Vorbild

Ein wichtiger Aspekt der Selbstorganisierenden Karte ist, dass die Neuronen auf der Wettbewerbsschicht in einer räumlichen Anordnung liegen, sodass sich für ein Lernverfahren nutzbare Nachbarschaftsbeziehungen ergeben. Selbstorganisierende Karten sind daher dem weichen Wettbewerbslernen zuzuordnen.

Ein Vorbild für die lokale Nähe von Neuronen die ähnliche Aufgaben erfüllen ist die somatomotorische Rinde im menschlichen Gehirn:

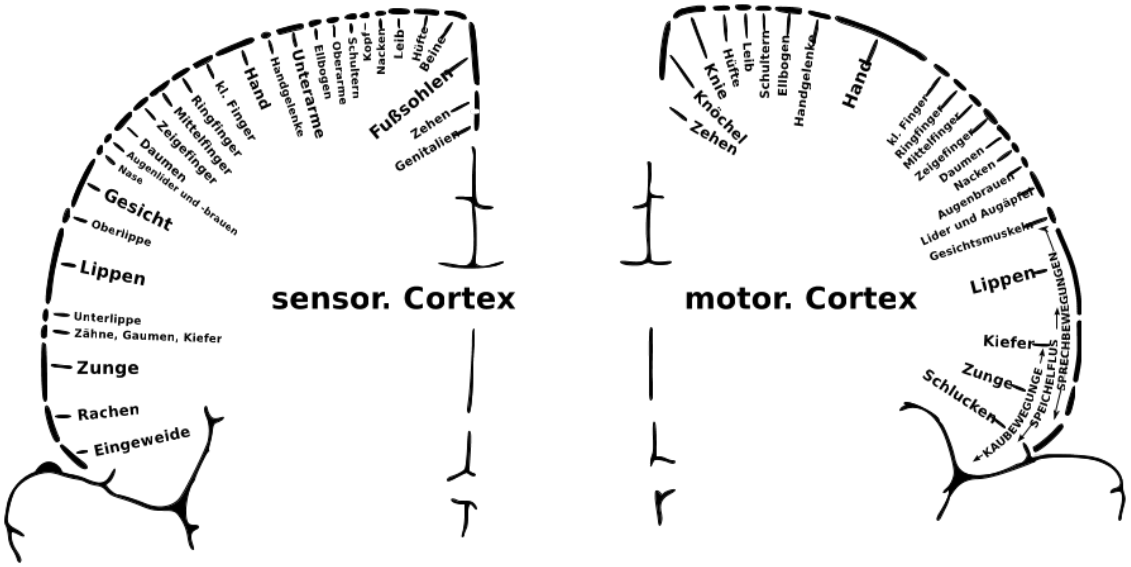

Abbildung 1: Sensomotorischer Cortex[1]

Dort liegen Bereiche die für unterschiedliche Teile des Körpers zuständig sind so beieinander das benachbarte Hirnregionen auch benachbarte Körperregionen steuern. Beispielsweise liegen die Bereiche zur Steuerung des Mundraums(Zunge, Lippen,...) dicht beeinander. Da eine solche Anordnung eine kartenartige Übertragung der Anatomie auf Nervenstrukturen darstellt, werden solche Bereiche als kortikale Karten bezeichnet. Dieses Prinzip wurde mit der Selbstorganisierenden Karte auch auf KNNs übertragen.

1.2.4 Grundidee der Selbstorganisierenden Karten

Die Lösungsidee der Selbstorganisierenden Karte zur Klassifikation von Daten besteht in der Ausnutzung des Prinzips der nahen räumlichen Anordnung ähnlicher Eingaben. Ähnliche Muster werden durch benachbarte Gewinner-Neuronen näher zusammen auf der Wettbewerbsschicht (der eigentlichen Karte) positioniert, sodass sich durch deren Häufung unterschiedliche Klassen von Mustern herausbilden. Im Laufe des Trainings entwickelt sich die Gewichtung der Verbindungen zwischen Eingabe und Wettbewerbsschicht so, dass ein Neuron als sog. Erregungszentrum die Ausgabe aller Eingabemuster einer bestimmten Klasse übernimmt und damit zu einem Repräsentanten einer Klasse von Daten wird.[5]

2 Architektur und Arbeitsweise

2.1 Aufbau des Neuronalen Netzes

Das neuronale Netz der Selbstorganisierenden Karte besteht aus 2 Schichten:

2.1.1 Eingabeschicht

Die Eingabeschicht beinhaltet die Eingabeneuronen, welche jeweils Verbindungen zu allen Neuronen der Kartenschicht besitzen, also vollständig vernetzt mit der Kartenschicht sind. Diese Gewichte legen das klassifizierende Verhalten des Netzes fest und ihre Anpassung ist Gegenstand des Lernverfahrens.

2.1.2 Kartenschicht

Die Kartenschicht ist die Wettbewerbsschicht bei der jedes Neuron j die Aktivität aller Eingabeneuronen $o_1, ..., o_k$ als Eingabe über Verbindungen mit den Gewichten $w_{1j}, ..., w_{kj}$ erhählt. Zusätzlich sind auch die Neuronen der Kartenschicht miteinander vollständig vernetzt. Zur Realisierung der räumlichen Anordnung besitzt die Kartenschicht eine bestimmte räumuliche Struktur. Häufig sind die Neuronen zur Anschaulichkeit in einem zweidimensionalen quadratischen Feld angeordnet. Denkbar sind aber auch andere Anordnungen. Die Neuronen könnten z.B. auch in eimem kreisförmigen Feld oder mehrdimensionalen Räumen angeordnet werden.[5]

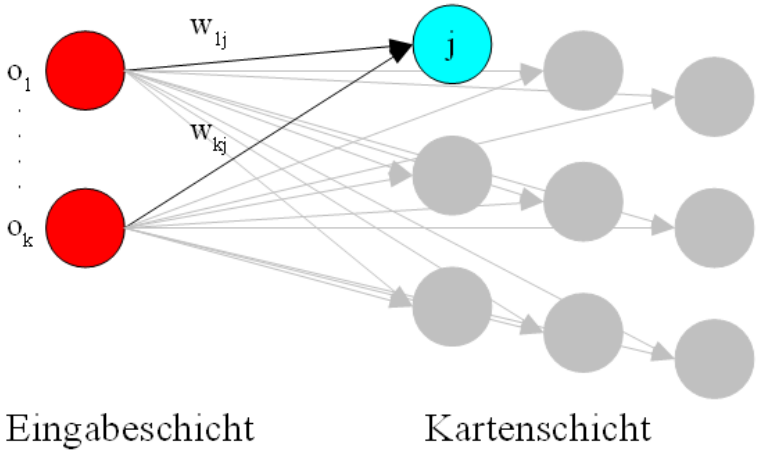

Abbildung 2: Schichtstruktur einer Selbstorganisierenden Karte

Aufgrund der räumlichen Anordnung gibt es eine Funktion $dist(x, y)$, welche ein Maß für den Abstand zweier Neuronen x und y darstellt. Um die räumliche Anordnung umzusetzen,

werden die Gewichte zwischen den Neuronen der Kartenschicht in Abhängikeit ihres Abstandes festgelegt. Dabei sollen sich benachbarte Neuronen erregen und entfernte Neuronen hemmen.[4]

2.2 Lernverfahren der Selbstorganisierenden Karte

2.2.1 Initialisierung

Zu Beginn des Lernvorgangs werden alle Verbindungsgewichte zwischen Eingabe- und Kartenschicht mit Zufallswerten initialisiert. Dies ist notwendig, da die Kartenschicht ansonsten bereits eine Information über eine Klassifikation enthielte, die noch nicht vorhanden sein kann.

2.2.2 Ablauf des Trainingsprozesses

Nun werden nacheinander zufällig Eingabemuster angelegt, das Gewinnerneuron bestimmt und die Gewichte angepasst bis eine bestimmte Anzahl an Durchläufen erreicht ist.

Dabei soll ein Lernkoeffzient λ dafür sorgen, das mit der Anzahl der Durchläufe auch der Einfluss nachfolgender Muster verringert wird, um so einen stabilen Zustand zu erreichen. λ wird daher nach jedem Muster verringert und konvergiert gegen 0.

Der Radius r gibt den Abstand an, bis zu dem ein vom Gewinner-Neuron entferntes Neuron bei der Anpassung der Gewichte berücksichtigt wird. Dieser soll ebenfalls während des Lernvorgangs verringert werden und gegen 0 konvergieren, um eine Verfeinerung der Selbstorganisierenden Karte zu ermöglichen.[6]

2.2.3 Ermittlung des Gewinner-Neurons

Ein Eingabemuster m_p bestehend aus den Werten $m_{p1}...m_{pk}$ lässt sich als Vektor $(m_{p1}, ..., m_{pk})$ darstellen. Dieser ist identisch zum Vektor $(o_1, ..., o_k)$, welcher die vom Eingabemuster hervorgerufene Erregung der Eingabeschicht darstellt. Für jedes Neuron der Kartenschicht j lassen sich die Gewichte der Verbindungen von der Eingabeschicht ebenso zu einem Vektor $W_j = (w_{1j}, ..., w_{kj})$ zusammenfassen.

Mithilfe der gebildeten Vektoren kann nun das Gewinner-Neuron z ermittelt werden. Hierzu werden zwei Verfahren eingesetzt[5]:

- Maximales Skalarprodukt:

  Hierbei wird das Neuron als Gewinner ermittelt, welches das maximale Skalarprodukt aus dem Vektor des Eingabemusters und dem Vektor der Verbindungsgewichte besitzt. Für z muss daher gelten: $\sum_i m_{pi} \cdot w_{iz} = \max_j \sum_i m_{pi} \cdot w_{ij}$

- Minimaler euklidischer Abstand:

  Hierbei ist das Gewinner-Neuron, das Neuron dessen Gewichtsvektor den geringsten Abtsand bezüglich der Euklidischen Norm zum Eingabevektor hat. Für z gilt: $\sum_i (m_{pi} - w_{iz})^2 = \min_j \sum_i (m_{pi} - w_{ij})^2$

Für die meisten Anwendungen ist es plausibler das Gewinner-Neuron so zu bestimmen, dass es am nähesten an der Eingabe liegt. Damit wird meist der euklidische Abstand gewählt.[6]

2.2.4 Anpassung des Neuronalen Netzes

Nach der Bestimmung des Gewinnerneurons werden die Gewichte des Neurons und seiner Nachbarschaft so verändert, dass sie eine noch stärkere Erregung durch das Eingabemuster erhalten. Dazu wird entsprechend die Differenz zwischen Vektor der Verbindungsgewichte und Vektor des Eingabemusters $m_i - w_{ij}$ für jedes Eingabeneuron i und Kartenneuron j gewählt und an die Gewichte der Eingabe entsprechend angenähert. Dabei soll bei der Ändrung die Lernrate λ und der Radius r berücksichtigt werden. Zusätzlich soll mit zunehmender Entfernung des Neurons j vom Gewinnerneuron z die Anpassung abnehmen, wozu ein Faktor h_{jz} verwendet wird. Frr das neue Gewicht w'_{ij} ergibt sich mit Berücksichtigung der Faktoren[6]:

$$w'_{ij} = \begin{cases} w_{ij} + \lambda \cdot h_{jz} \cdot (m_i - w_{ij}), & \text{falls } dist(j,z) \leq r \\ w_{ij}, & \text{sonst} \end{cases}$$

Für die Bestimmung des Faktors h_{jz} wird eine Nachbarschaftsfunktion bestimmt, welche die Eigenschaft haben muss, bei zunehmendem Abstand der Vektoren $dist(j,z)$ abzunehmen.

Häufig wird die Gauß'sche Glockenkurve verwendet.

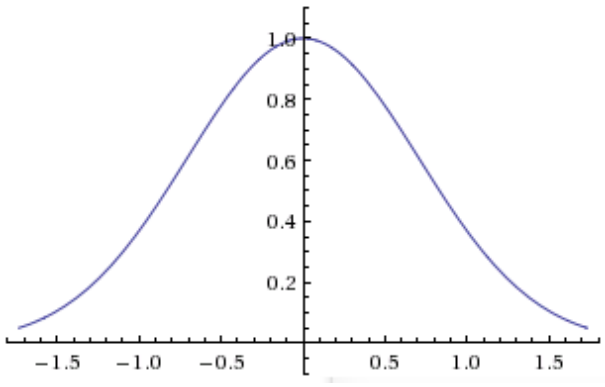

Abbildung 3: Gauß'sche Glockenkurve

Um die Glockenkurve bei zunehmenden Lernfortschritt flacher verlaufen zu lassen, kann der Radius r zusattzlich als Paramter verwendet werden sodass sich $h_{jz} = e^{-\frac{(dist(j,z)^2)}{(2 \cdot r^2)}}$ ergibt.

3 Anwendungen

3.1 Problem des Handlungsreisenden

Eine Anwendungsmöglichkeit für Selbstorganisierende Karten ist das Problem des Handlungsreisenden. Hierbei soll zu einer gegebenen Menge an Orten die kürzeste Rundreiseroute gefunden werden. Hierbei kann mit Selbstorganisierenden Karten im Allgemeinen nur eine suboptimale Lösung erreicht werden.

Bei einem üblicherweise zweidimensionalen Koordinatensystem, in welchem die Orte dargstellt sind, besteht die Eingabeschicht aus zwei Neuronen, die jeweils die x- und y-Koordinate repräsentieren. Die Kartenschicht besteht aus mindestens so vielen Neuronen, wie Orte auf der Karte vorhanden sind. Die Kartenschicht ist eine eindimensionale Kette, sodass die Kartenneuronen einen Vorgänger und Nachfolger besitzen. Bei der Initialisierung der Gewichte zwischen Eingabe- und Kartenschicht werden diese so gewählt, dass die Kartenneuronen visualisiert einen Kreis in der Mitte des Koordinatensystems ergeben. Eine zufällige Initialisierung würde hierbei zu einer Verteilung entgegen der Nachbarschaftsbeziehungen führen.

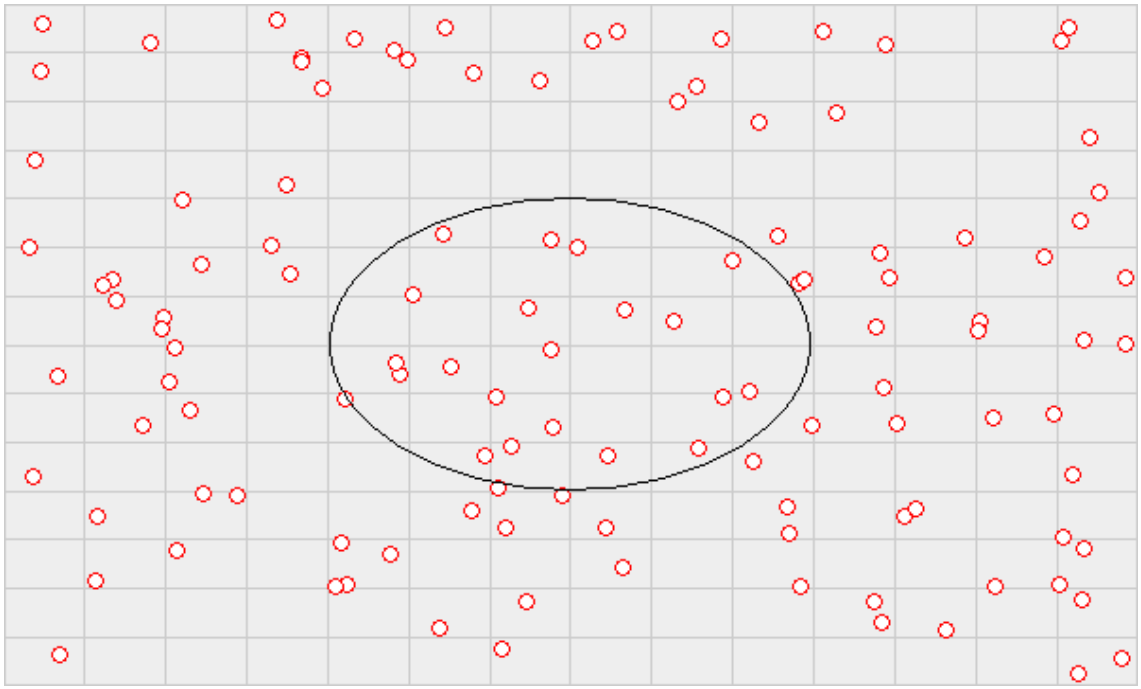

Abbildung 4: Initialisierung einer SOM für das Problem des Handlungsreisenden

Anschließend wird das Lernverfahren der Selbstorganisierende Karte angewandt, sodass sich einzelne Kartenneuronen als Gewinner-Neuron bestimmter Orte ergeben und sich Ihnen annähern. Nun muss in einem weiteren Schritt den Orten die entsprechenden Kartenneuronen zugewiesen werden. Die generierte Route lässt sich nun anhand der in der

Kartenschicht festgelegten Reihenfolge der Kartenneuronen mit den zugehörigen Orten ablesen.[6]

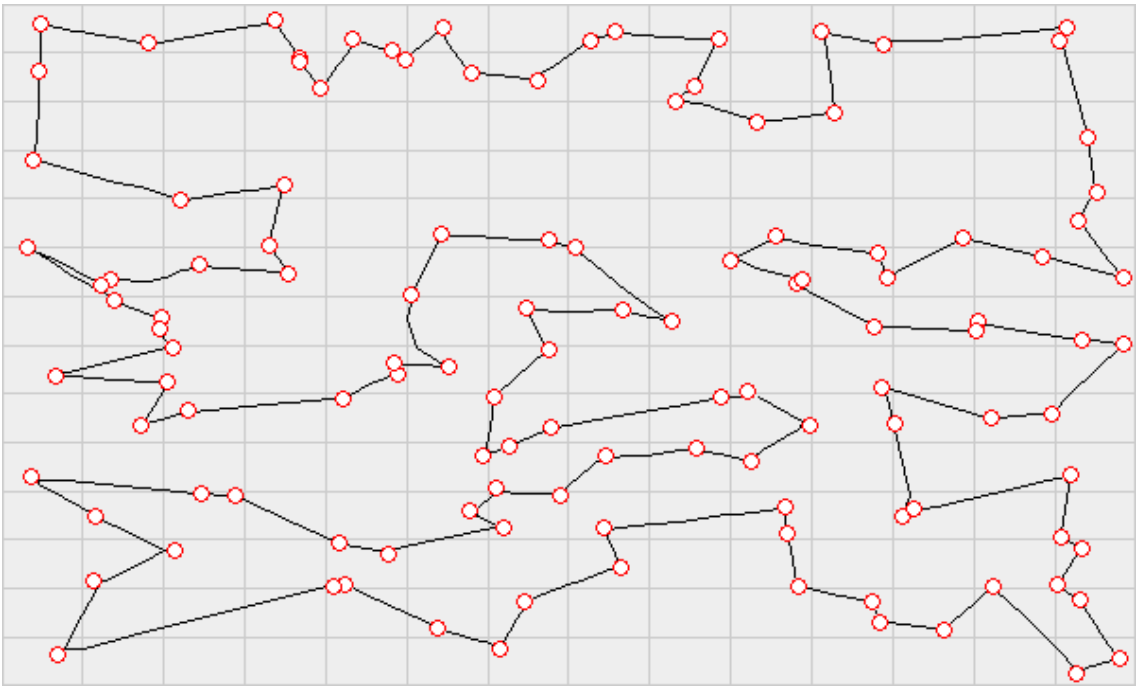

Abbildung 5: Durch SOM generierte Route[2]

Literatur

[1] *[URL]http://de.wikipedia.org/wiki/Somatotopie.*

[2] *[URL]http://www2.htw-dresden.de/ iwe/Belege/Boerner/.*

[3] K. Berns and T. Kolb. *Neuronale Netze für technische Anwendungen.* Springer, 1st edition, 1994.

[4] N. Hoffmann. *Kleines Handbuch Neuronale Netze Anwendungsorientiertes Wissen zum Nachschlagen.* Vieweg, 1st edition, 1993.

[5] U. Lämmel and J. Cleve. *Lehr und Übungsbuch Künstliche Intelligenz.* Fachbuchverlag Leipzig, 1st edition, 2001.

[6] U. Lämmel and J. Cleve. *Künstliche Intelligenz.* Hanser, 3rd edition, 2008.