

BrainDesigner

Anleitung

Version 0.02 – 2. Juli 2013

Labor für Neurorobotik

Humboldt-Universität zu Berlin
Institut für Informatik, Labor für Neurorobotik
Dr. Manfred Hild

Unter den Linden 6
10099 Berlin, Deutschland



Inhaltsverzeichnis

Inhaltsverzeichnis	1
Einführung	2
Programm-Übersicht	3
Bibliotheks-/Projektbereich	3
Kontextmenü Bibliothek/Projekt	5
Kontextmenü Elemente	5
Bearbeitungsbereich	5
Quellcode-Editor	5
Grafischer Editor	6
Cockpit	7
Curve Display	8
Beispiel	10
Units und Synapses	12
Inputs, Outputs, Parameters, Internals	12
ByteCode	12
Plugins	14

Einführung

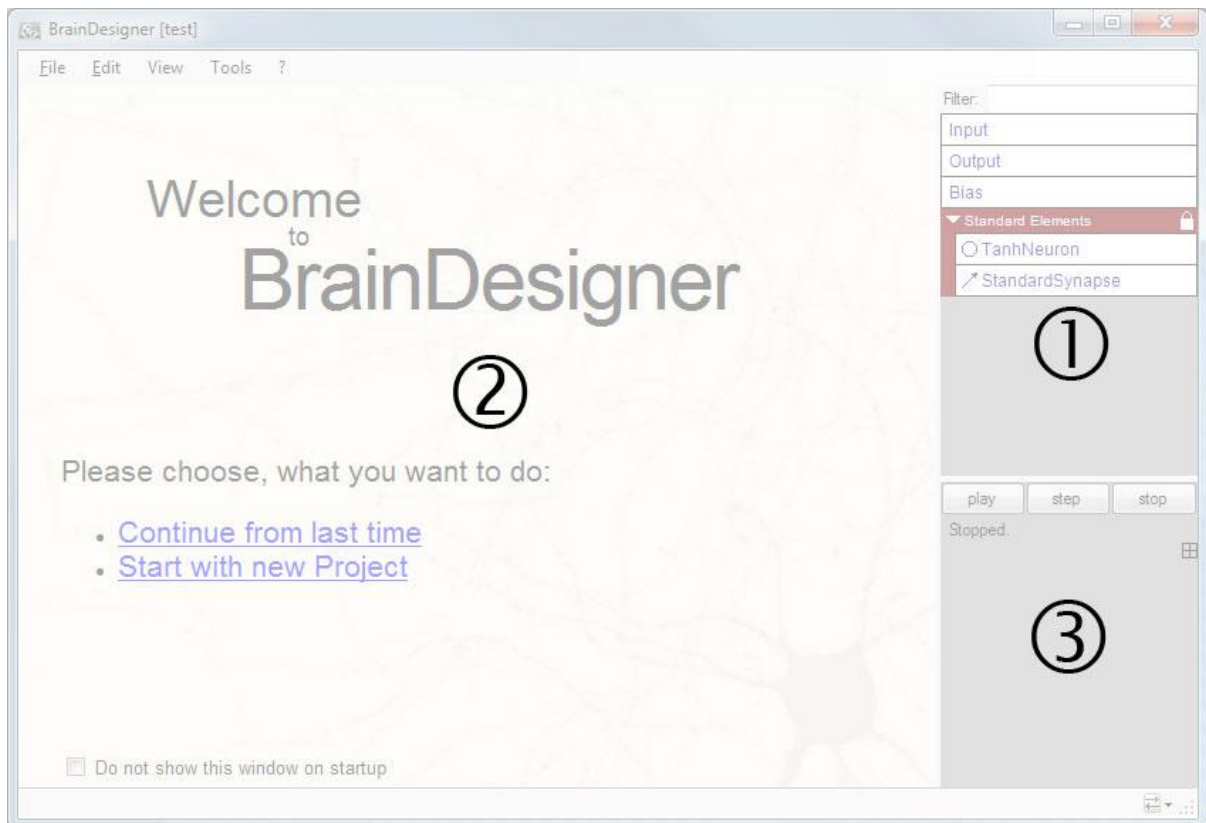
BrainDesigner ist ein Programm, um künstliche Neuronale Netze grafisch zu erstellen. Hinter Neuronen und Synapsen liegt dabei ausführbarer Code, der ermöglicht, verschiedene Berechnungsvorschriften und Lernverfahren zu implementieren. Grafisch zusammengestellte Netze können wiederum in weitere Netze eingebunden werden, wodurch Netz-Hierarchien erstellt werden können.

Über Plug-Ins kann nicht nur auf eigene Befehle im Code reagiert werden, es können auch beliebige In- und Outputs definiert werden und somit an den PC angeschlossene Roboter oder andere Programme gesteuert werden.

Programm-Übersicht

Nach dem erstmaligen Start des BrainDesigners teilt sich das Programm in drei Bereiche:

1. Bibliotheks-/Projektbereich
2. Bearbeitungsbereich
3. Cockpit



Im Bibliotheks-/Projektbereich finden sich alle verfügbaren Bibliotheken und Projekte mit ihren Elementen. Diese Elemente können im Bearbeitungsbereich bearbeitet werden. Für Neuronen oder Synapsen kann entsprechend Code bearbeitet werden, für ein neuronales Netz gibt es eine grafische Bearbeitungsfläche. Im Cockpit kann das Abspielen des Netzes gesteuert werden und es können dynamisch anpassbare Parameter eingestellt werden.

Bibliotheks-/Projektbereich

Im Bibliotheks-/Projektbereich werden alle Elemente von eingebundenen Bibliotheken und Projekten angezeigt. Ganz oben werden dabei die Systemelemente „Input“, „Output“ und „Bias“ angezeigt. Darunter folgen Bibliotheken und Projekte. Zu einer Bibliothek oder einem Projekt gehörende Elemente werden eingerückt dargestellt.

Bibliotheken sind dunkelrot hinterlegt, ein gerade aktives Projekt dunkelblau, alle anderen Projekte grau. Ein Schloss-Symbol hinter einem Projekt/einer Bibliothek indiziert, dass dieses „gelockt“ ist. Es können in diesem Fall weder Elemente verändert werden, noch neue hinzugefügt werden. Dies sollte vor allem auf Bibliotheken zutreffen.

Im Menü unter „File“ -> „New Project...“ oder „File“ -> „Add existing Project...“ können Projekte hinzugefügt werden.

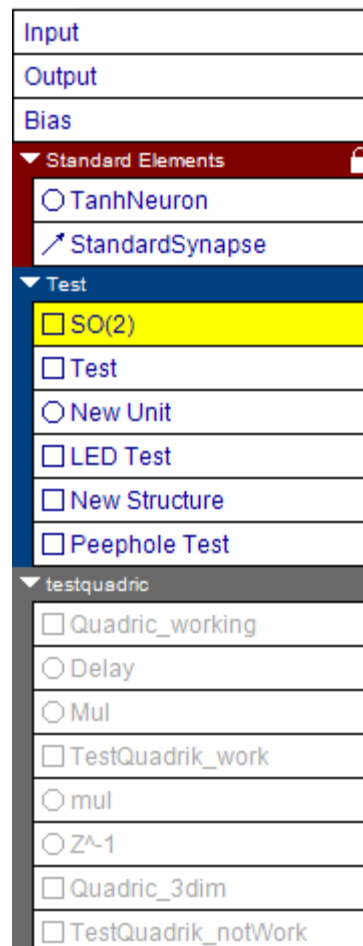


Abbildung 1: Beispielhafter Bibliotheks-/Projektbereich

Elemente innerhalb einer Bibliothek oder eines Projekts können sein:

„Units“: Dies sind Neuronen bzw. andere Berechnungseinheiten, die im grafischen neuronalen Netz Knoten entsprechen. Im Bibliotheks-/Projektbereich werden diese mit einem Kreis vor dem Namen dargestellt.

„Synapses“: Dies sind Synapsen, also Kanten im neuronalen Netz. Diese werden im Bibliotheks-/Projektbereich mit einem Pfeilsymbol dargestellt.

„Structures“: Dies sind grafische neuronale Netze. Sie werden im Bibliotheks-/Projektbereich mit einem Quadrat vor dem Namen dargestellt.

Das gerade im Bearbeitungsbereich geöffnete Element wird gelb unterlegt dargestellt. Bei veränderten und noch nicht gespeicherten Elementen wird rechts ein rotes Quadrat abgebildet. Elemente werden mit einem Doppelklick geöffnet. Ein einfacher Linksklick auf den Titel einer Bibliothek bzw. eines Projekts klappt dieses ein, ein weiterer Linksklick wieder aus.

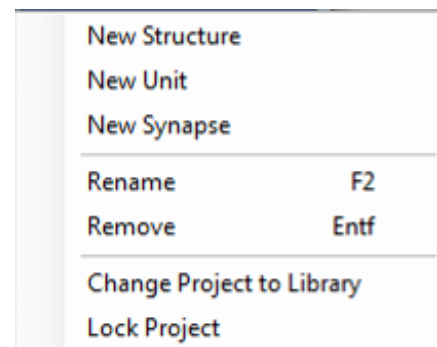
Per Drag&Drop kann die Reihenfolge der Elemente geändert werden bzw. es können Elemente aus einem Projekt in ein anderes importiert werden.

In der mitgelieferten Standardbibliothek findet sich eine Unit „TanhNeuron“: ein Neuron mit einem Ein- und Ausgang und der Transferfunktion Tangens Hyperbolicus. Am Eingang werden beliebig viele angeschlossene Synapsen aufaddiert. Der Synapsentyp „StandardSynapse“ hat einen Parameter („w“), mit dem der Eingangswert multipliziert wird. Mit diesen beiden Elementen lassen sich einfache neuronale Netze realisieren.

Sowohl Projekte als auch Elemente bieten per Rechtsklick ein Kontextmenü mit weiteren Möglichkeiten.

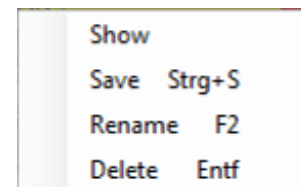
Kontextmenü Bibliothek/Projekt

Im Kontextmenü einer Bibliothek oder eines Projekts gibt es zunächst die Möglichkeit, neue Structures, Units und Synapsen anzulegen. Des Weiteren kann das Projekt umbenannt werden bzw. aus dem Bibliotheks-/Projektbereich entfernt werden. Hier ist es auch möglich, ein Projekt in eine Bibliothek bzw. umgekehrt umzuwandeln und das Projekt/die Bibliothek zu sperren.



Kontextmenü Elemente

Im Kontextmenü von Elementen können diese geöffnet („Show“), gespeichert („Save“), umbenannt („Rename“) oder gelöscht („Delete“) werden.



Bearbeitungsbereich

Im Bearbeitungsbereich wird je nach geöffnetem Elementtyp entweder ein Quellcode-Editor oder ein grafischer Editor angezeigt. Beim Start (oder per Menü: „View“ -> „Show Start Screen“) wird hier ein Startfenster angezeigt, über das ein neues Projekt gestartet werden kann oder die Arbeit fortgesetzt werden kann.

Quellcode-Editor

Im Quellcode-Editor gibt es zusätzlich zu einem Texteditor mit Zeilennummern und Syntax Highlighting die Möglichkeit, Ein- und Ausgänge, Parameter und interne Variablen („Internals“) für dieses Element zu definieren. Auf die Erstellung von Units und Synapsen wird detaillierter in Kapitel 0 eingegangen.

Der Code kann im Menü („File“ -> „Export“ -> „ByteCode“) als Textdatei oder als RTF-Datei (Rich Text Format) mit Syntax Highlighting exportiert werden.

Grafischer Editor

Herzstück des BrainDesigners ist der grafische Editor. Wird eine neu angelegte Structure geöffnet, bietet sich ein leerer Zeichenbereich. Neue Instanzen von Elementen fügt man per Drag&Drop aus dem Bibliotheks-/Projektbereich hinzu. Elemente aus dem aktiven Projekt oder aus Bibliotheken werden dabei einfach eingebunden, Elemente aus inaktiven Projekten werden vorher in das aktive Projekt importiert. Um ein Projekt an einem anderen Rechner öffnen zu können benötigt man neben dem aktiven Projekt also auch alle Bibliotheken, aus denen Elemente verwendet werden.

Mit einfachem Klick im grafischen Editor markiert man Elemente, mit Strg+Klick kann man weitere Elemente zur Auswahl hinzufügen. Per Klick auf einen freien Bereich und ziehen mit der Maus kann man einen ganzen Bereich markieren.

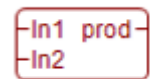
Elemente werden normalerweise in einem dunklen braun dargestellt, markierte Elemente werden blau dargestellt. Die sich an der aktuellen Mauszeigerposition befindliche Elementinstanz wird grün dargestellt, zusätzlich wird das Element im Bibliotheks-/Projektbereich grün unterlegt. Fährt man mit der Maus im Bibliotheks-/Projektbereich über ein Element werden im grafischen Editor alle Instanzen grün dargestellt.

Mit den Tastenkombinationen Strg+Plus und Strg+Minus kann die Anzeige gezoomt werden. Zusätzlich stehen im Menü unter „View“ -> „Zoom“ weitere Zoomoptionen zur Verfügung.

Die grafische Darstellung kann im Menü („File“ -> „Export“ -> „Structure Graphics“) als PNG (Pixelgrafik) oder SVG (Vektorgrafik) exportiert werden.

Units

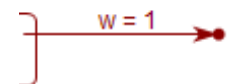
Units werden je nach Konfiguration entweder als Kreis dargestellt (bei jeweils einem Ein- und Ausgang) oder als Rechteck mit abgerundeten Ecken und benannt dargestellten Ein- und Ausgängen. Das Beispiel rechts zeigt eine Unit mit zwei Eingängen („In1“ und „In2“) und einem Ausgang „prod“. Synapsen können nur an diesen Ein- oder Ausgängen angeschlossen werden. Bei einfachen Units (ohne benannte Ein- und Ausgänge) gibt es acht Anschlusspunkte auf dem Kreis, die alle äquivalent sind. Units können per Drag&Drop verschoben werden.



Ein- und

Synapses

Synapses werden als Pfeile dargestellt. Das Beispiel rechts ist am Eingang an eine Unit angeschlossen, der Ausgang (das Pfeilende) ist offen. Offene Ein- und Ausgänge werden mit einem kleinen Kreis markiert, um fälschlich unangeschlossene Synapsen besser zu erkennen. Auf einer Synapse werden eventuelle Parameter dargestellt (im Beispiel ist der Parameter „w“ auf „1“ gesetzt). Über ein Kontextmenü kann die Darstellung der Parameter für jede Synapsen-Instanz eingestellt werden. So können unwichtige Parameter in der Darstellung ausgeblendet werden.



Synapses können nicht als Ganzes verschoben werden. Wird ein Objekt verschoben, an das eine Synapse angeschlossen ist, wird das jeweilige Ende mitverschoben. Jeweils der Start- und Endpunkt einer Synapse können manuell verschoben werden. Schiebt man einen solchen Punkt in die Nähe eines Anschlusspunkts einer Unit oder einer Structure, wird die Synapse angeschlossen.

Auch die Mitte einer Synapse ist sensitiv. Wird diese angeklickt und verschoben, wird die Krümmung der Kurve eingestellt. Wird die Maus nach links oder rechts (relativ zum Start- und Endpunkt) bewegt, wird die Breite der Krümmung geändert.

Structures

Structures, die in eine andere Structure eingefügt werden, werden als Rechtecke dargestellt. Ähnlich wie bei Units werden dabei Ein- und Ausgänge dargestellt. Die Ein- und Ausgänge definieren sich durch die innerhalb der eingebundenen Structure vorhandenen In- und Outputs („Normal Input“ und „Normal Output“).

Structures können per Drag&Drop verschoben werden.

Systemelemente

Die Systemelemente Bias, Input und Output werden jeweils als Halbkreis dargestellt, wobei Bias und Input nach links offen sind, Output nach rechts.

Neben „Normal Inputs“ und „Normal Outputs“, die in einzubettenden Structures benutzt werden, gibt es je nach verwendeten Plugins verschiedene Roboter-In- und Outputs.

Parameter

Alle Elemente können verschiedene Parameter haben. Bei Units/Synapses sind dies normalerweise Berechnungsparameter (z.B. das Gewicht „w“ bei einer StandardSynapse), bei Inputs/Outputs ist das die Eingabe-Art (z.B. „Normal Input“ oder „File Input“). Werden im grafischen Editor Elemente selektiert, werden sämtliche möglichen Parameter in einem Parameterbereich unterhalb der Zeichenfläche dargestellt und können eingestellt werden.

Parameter von Units/Synapses werden dabei in der Structures-Hierarchie nach oben durchgegeben: Wird beispielsweise der Parameter „w“ einer Synapse in Structure 1 nicht mit einem konkreten Wert sondern mit „x“ festgelegt und diese Structure in eine andere Structure (Structure 2) eingebettet, hat die eingebettete Structure 1 automatisch den externen Parameter „x“, der auf Ebene der Structure 2 mit einem Wert festgelegt werden kann. So kann ein und dieselbe Structure mehrfach in eine andere Structure eingefügt werden, aber jeweils andere Parameter bekommen.

Parameter für Units/Synapses können nicht nur mit Werten („1“, „-0.2“) und externen Parameternamen („x“) belegt werden, auch einfache Berechnungen sind möglich. So kann der externe Parameter „x“ in mehreren Synapsen vorkommen, beispielsweise als „w = x“, „w = -x“ und „w = 3 * x“.

Solange eine Structure nicht in eine andere eingebunden wird, können externe Parameter auch über „Slider“ im Cockpit eingestellt werden, siehe Seite 8.

Cockpit

Im Cockpit stehen im oberen Teil Schaltflächen zur Steuerung des Netzes zur Verfügung. Es gibt dabei mindestens die Schaltflächen „Play“, „Step“ und „Stop“. Mit „Play“ wird das Netz gestartet, der Button verwandelt sich dann in einen „Pause“-Button. Mit „Step“ wird genau ein Zeitschritt berechnet, mit „Stop“ die Berechnung abgebrochen. Unterhalb der Buttons wird der aktuelle Status angezeigt.

Die Frequenz der Berechnung (Anzahl Zeitschritte pro Sekunde) kann dabei entweder von einem externen Roboter vorgegeben werden oder intern eingestellt werden, je nach aktivierten Plugins. Wenn kein aktives Plugin die Frequenz vorgibt, kann diese im Menü unter „Tools“ -> „Settings“ eingestellt werden, Standardwert ist 100 Hz. Dort gibt es auch die Möglichkeit, auf „Maximum Speed“ zu stellen, dann gibt die Berechnungsgeschwindigkeit des PCs die Frequenz vor.

Manche Plugins führen dazu, dass vor dem „Play“-Button noch ein „Init“-Button angezeigt wird. In diesem Fall werden die Initialisierung des Netzes und der Start der Berechnung getrennt. Dies ist beispielsweise bei Robotern nötig, bei denen aus dem neuronalen Netz ausführbarer Code kompiliert wird, der zunächst auf die Roboter transferiert werden muss. Nach diesem Schritt kann man dann beliebig oft „Play“ und „Stop“ ausführen, ohne erneut den zeitintensiven Schritt des Kompilierens und Deployens ausführen zu müssen.

Unterhalb der Steuerungseinheit gibt es einen Bereich, in dem beliebig viele „Slider“ hinzugefügt werden können (mittels eines Buttons mit einem Plus-Zeichen rechts oben). In einem Slider kann entweder per Tastatur oder per Maus ein Wert zwischen den änderbaren Minimal- und Maximalwerten eingestellt werden (rechts im Beispiel ein Wert zwischen -1 und +1, aktuell 0,3). Über dieser Einstellmöglichkeit befindet sich ein Feld, in dem der Parametername angegeben wird, für den dieser Wert gilt. Wird dort wie im Beispiel „x“ eingetragen und an einer Stelle im neuronalen Netz ein Synapsengewicht „w“ auf „x“ gestellt, wird bei der Berechnung der aktuelle Wert des Sliders verwendet, ist das Synapsengewicht auf „ $1 - x$ “ gestellt, wird diese Berechnung durchgeführt. So kann schnell während ein Netz läuft, ein Parameter geändert werden und das geänderte Verhalten beobachtet werden.



Curve Display

Oberhalb des Bearbeitungsbereichs kann optional das Curve Display angezeigt werden. Manuell kann diese Darstellung im Menü unter „View“->„Show Curves“ an- und ausgeschaltet werden.

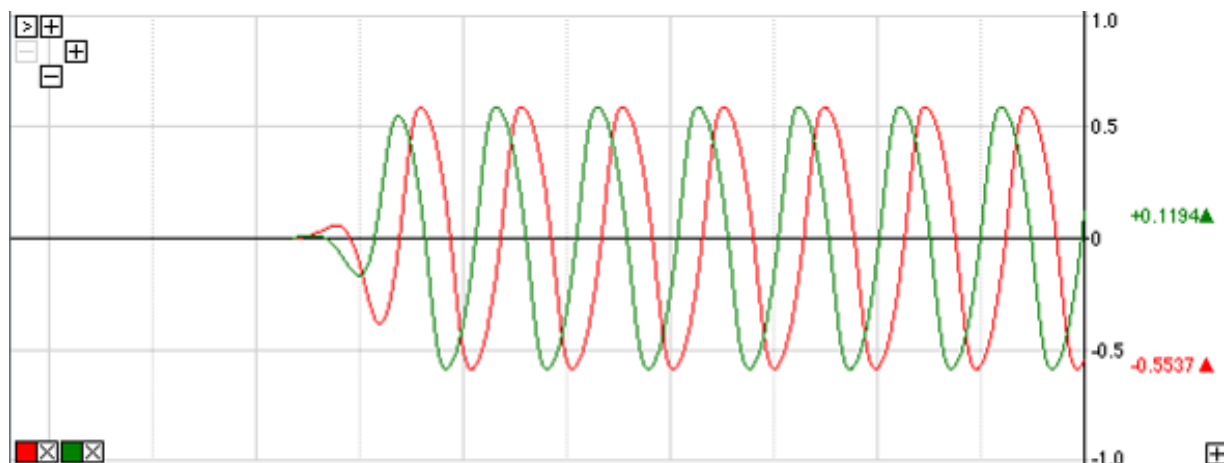


Abbildung 2: Ein beispielhaftes Curve Display

Im Curve Display können beliebige Werte aus dem neuronalen Netz angezeigt werden, beispielsweise Neuronenausgangswerte oder Synapsenausgangswerte. Die Werte werden dabei grafisch dargestellt, wobei auf der x-Achse die Zeitschritte abgetragen sind, auf der y-Achse die Werte. Der ak-

tuellste Wert wird ganz rechts dargestellt, die Darstellung verschiebt sich während des Abspielens nach links.

Im Beispiel in der Abbildung 2: Ein beispielhaftes Curve DisplayAbbildung 2 werden zwei Werte dargestellt (rot und grün). Links unten werden alle dargestellten Kurven abgebildet. Fährt man mit der Maus über eine der Legenden wird das dazugehörige Element im grafischen Editor hervorgehoben. Mit einem Klick auf das Farbquadrat kann man die Farbe ändern, mit einem Klick auf das X löscht man die entsprechende Kurve.

Kurven können mittels des Kontextmenüs der Elemente im grafischen Editor hinzugefügt werden. Wird mit rechts auf den Start- oder Endpunkt einer Synapse geklickt, erscheint ein Kontextmenü mit dem Eintrag „Show in Curve Display“. Auch über dieses Kontextmenü kann man die Kurve wieder entfernen.

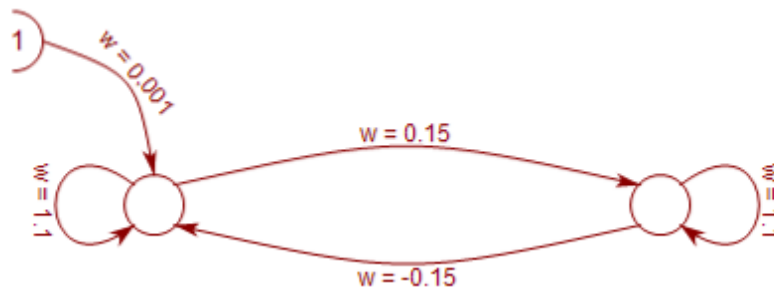
Mit den vier Plus- und Minus-Schaltflächen oben links kann sowohl die x- als auch die y-Achse gezoomt werden. Mit der Plus-Schaltfläche rechts unten können weitere Displays hinzugefügt werden, die dann unabhängig voneinander übereinander dargestellt werden. Per Drag&Drop mit dem zugehörigen Farbquadrat kann man eine Kurve in eine andere „Subcurve“ ziehen.

Ganz links oben befindet sich eine weitere Schaltfläche mit einem kleinen Pfeil nach rechts. Damit lässt sich ein 2D-Display einblenden. In diesem können zwei Werte gegeneinander abgetragen werden. Einer der Werte ist dabei auf der x-Achse, der andere auf der y-Achse. Dargestellt werden die letzten 500 Zeitschritte. Per Drag&Drop des zugehörigen Farbquadrats im normalen Curve Display auf die x- bzw. y-Achse werden diese zugeordnet.

Die Einstellungen des Curve Displays (welche Kurven, Zoomstufen etc.) gehören jeweils zu einer Struktur und werden mit dieser gespeichert.

Beispiel

In diesem Kapitel soll beispielhaft das Erstellen eines kleinen neuronalen Netzes mithilfe des BrainDesigners erläutert werden.



Das dargestellte neuronale Netz besteht aus zwei Neuronen, die vollständig verbunden sind (vier Synapsen). Zusätzlich gibt es einen Bias-Eingang, der auf das linke Neuron gegeben wird. Bei der dargestellten Parameterkonstellation schwingen die Werte der Neuronen. Dieses Netz wird als SO(2) bezeichnet.

Starten Sie den BrainDesigner und erstellen Sie ein neues Projekt (Menü: „File“ -> „New Project“). Wenn Sie nach dem Dateinamen gefragt werden, geben Sie `Beispiel.bdp` an. Dem neuen Projekt wurde automatisch eine neue Structure hinzugefügt und diese geöffnet. Klicken Sie mit rechts auf den Eintrag „New Structure“ unterhalb des Projekts „Beispiel“ im Bibliotheks-/Projektbereich und wählen Sie im Kontextmenü den Eintrag „Rename“. Benennen Sie das Element in „SO(2)“ um. Bestätigen Sie mit Enter.

Fügen Sie im grafischen Editor zwei Neuronen hinzu, indem Sie auf das Element „TanhNeuron“ im Bibliotheks-/Projektbereich klicken und es bei gedrückter Maustaste in den freien Zeichenbereich links ziehen. Ordnen Sie diese nebeneinander an, wie in der Grafik oben dargestellt.

Fügen Sie nun vier Synapsen hinzu. Auch diese ziehen Sie aus dem Bibliotheks-/Projektbereich herüber. Verbinden Sie diese wie in der beispielhaften Abbildung dargestellt. Per Klick auf den Start- bzw. Endpunkt einer Synapse verschieben Sie diesen. Sobald Start- bzw. Endpunkt mit einem Neuron verbunden sind, verschwinden die kleinen Kreise. Klicken Sie auf die Mitte einer Synapse und bewegen Sie die Maus bei gedrückter Maustaste um die Krümmung einer Synapse einzustellen.

Fügen Sie ein Element „Bias“ hinzu und eine weitere Synapse um die dargestellte Topologie zu erreichen.

Per einfachen Klick auf eine Synapse (den Start-, Mittel- oder Endpunkt) markieren Sie diese (sie wird nun blau dargestellt). Im unteren Bereich können Sie nun einen Wert für den Parameter „w“ einstellen. Setzen Sie w auf 0.001 für die Synapse vom Bias zum linken Neuron, 1.1 für die beiden Selbstkopplungen und 0.15 bzw. -0.15 für die Ringkopplungen.

Klicken Sie mit rechts auf das linke Neuron und wählen Sie im Kontextmenü „Show in Curve Display“. Das Curve Display wird automatisch eingeblendet.

Klicken Sie im Cockpit auf den Button „Play“. Sie sollten nun die Schwingung des linken Neurons im Curve Display sehen.

Lassen Sie sich zusätzlich die Werte des rechten Neurons anzeigen (Rechtsklick auf das rechte Neuron, im Kontextmenü „Show in Curve Display“ wählen). Dieses wird nun zusätzlich in grün dargestellt. Mit einem Klick links oben im Curve Display auf den kleinen nach rechts gerichteten Pfeil wird das 2D-Display eingeblendet. Ziehen Sie mit der Maus das rote Quadrat (Wert des linken Neurons) auf die x-Achse des 2D-Displays, das grüne Quadrat (Wert des rechten Neurons) auf die y-Achse. Sie sehen nun die 500 letzten Werte gegeneinander abgetragen.

Mit dem Cockpit-Button „Stop“ beenden Sie die Berechnung.

Fügen Sie nun ein Output-Element hinzu und verbinden Sie das rechte Neuron mittels einer Synapse mit dem Gewicht $w = 1$ mit diesem Output. Markieren Sie den Output und ändern Sie den Parameter „Name“ auf „osc“ wie „oscillator“. Ändern Sie die Ringkopplungen von „0.15“ und „-0.15“ auf „x“ und „-x“.

Fügen Sie in das Projekt „Beispiel“ eine neue Structure ein (Rechtsklick auf den Projekttitel, „New Structure“ im Menü). Die neue Structure wird mit einer leeren Zeichenfläche geöffnet. Ziehen Sie eine Instanz der gerade erstellen Structure „SO(2)“ in die Zeichenfläche. Diese wird als Rechteck mit einem Ausgang „osc“ dargestellt.

Markieren Sie die eingefügte Structure. Im Parameterbereich sollte ein Parameter „x“ dargestellt werden, der zunächst auch den Wert „x“ hat. Ändern Sie diesen auf „0.15“.

Lassen Sie sich den Ausgang der eingebetteten Structure anzeigen, indem Sie mit rechts auf diese klicken und im Kontextmenü „Show in Curve Display“ -> „osc“ wählen.

Starten Sie nun erneut den Berechnungsvorgang indem Sie im Cockpit auf „Play“ klicken. Sie sollten die gleiche Schwingung sehen wie vorher. Sie könnten nun weitere SO(2)-Instanzen hinzufügen und sich die Schwingungen für verschiedene Ringkopplungen anschauen.

Units und Synapses

Units und Synapses sind die zentralen Berechnungseinheiten hinter einem grafischen neuronalen Netz. Ihnen liegt jeweils einfacher Code zugrunde. In diesem Kapitel soll erläutert werden, wie eigene Unit- und Synapses-Typen erstellt werden können.

Units und Synapses werden im grafischen Editor anders dargestellt, sind aber im Prinzip das gleiche: Berechnungseinheiten mit Ein- und Ausgängen, Parametern und internen Variablen („Internals“). Synapses sind dabei der Spezialfall und haben immer einen Ein- und Ausgang. Alle Synapses, die an einem Ein- oder Ausgang einer Unit oder einer Structure angeschlossen werden, werden automatisch addiert.

Inputs, Outputs, Parameters, Internals

Wird der ByteCode-Editor für eine Unit aufgerufen, kann im unteren Bereich eingegeben werden, welche Inputs, Outputs, Parameters und Internals diese Unit haben soll. Bei Synapses fehlt die Möglichkeit, Inputs und Outputs anzugeben, diese haben automatisch einen Input mit dem Namen „Input“ und einen Output mit dem Namen „Output“. Wird bei einer Unit weder ein benannter Input noch ein benannter Output angegeben, stehen auch implizit jeweils ein Input und ein Output mit diesen Namen bereit. Nur in diesem Fall wird die vereinfachte Neuronendarstellung (als Kreis) im grafischen Editor gewählt. Sind benannte Inputs oder Outputs vorhanden wird die Unit als Rechteck mit abgerundeten Ecken und benannten In- und Outputs dargestellt. Im Bytecode werden In- und Outputs mit ihrem Namen angesprochen.

Parameter sind Werte, die auf Ebene des grafischen neuronalen Netzes für jede Instanz vergeben werden können, sich aber über die Zeit nicht ändern. Für Parameter kann ein Standardwert angegeben werden. Bei Units werden Parameter im grafischen Editor nicht dargestellt, bei Synapses werden die Parameter auf dem Pfeil angezeigt.

Internals sind Werte, die zur Berechnung benutzt werden und über die Zeit erhalten bleiben, aber nicht nach außen gegeben werden. Sie unterscheiden sich von Outputs praktisch nur durch diese Tatsache.

ByteCode

Der ByteCode richtet sich nach einigen Regeln. Die größte Struktur ist eine Zeitstruktur. Für jeden Codeblock muss angegeben werden, zu welcher Zeit relativ zu anderen Codeblöcken (auch anderer Units) der Code ausgeführt wird. Dafür werden Zeitstempel zwischen 100 und 1000 genutzt. Steht in einer Zeile beispielsweise „200:“ (der Zeitstempel 200 gefolgt von einem Doppelpunkt) werden alle folgenden Zeilen zum Zeitpunkt „200“ sequentiell ausgeführt (bis zum Ende des Codes oder bis eine weitere Zeile Berechnungen zu einer anderen Zeit definiert).

Gibt es mehrere Code-Blöcke zu einem Zeitstempel (sei es durch mehrere Instanzen einer Unit oder durch mehrere Units mit dem gleichen Zeitstempel) ist die Ausführungsreihenfolge nicht definiert.

Bei den Beispiелеlementen „TanhNeuron“ und „StandardSynapse“ wird der Code der Synapse immer vor dem des Neurons ausgeführt („200“ und „300“), damit es entsprechend der Theorie künstlicher Neuroner Netze nur bei Neuronen zu einer Zeitverzögerung kommt.

Der ByteCode an sich ist einfach gehalten. Jede Codezeile beginnt mit einem Befehl, gefolgt von beliebigen Parametern, die durch Kommata oder Leerzeichen getrennt werden.

Der Befehlssatz ist abhängig von den aktivierten Berechnungs-Plugins. So kann ein Plugin, das ausführbaren Code für einen Roboter erstellt, nur einen eingeschränkten Befehlssatz haben, denkbar ist aber auch ein Plugin, das sehr viele Befehle zur Verfügung stellt.

Das im BrainDesigner integrierte und standardmäßig aktive Plugin „Standard“ enthält die Routine „Calculate“, die einige einfache Befehle zur Berechnung auf dem PC bereitstellt. Die Befehle sind bewusst einfach gehalten und sollten auch auf fast jedem Robotersystem umsetzbar sein. In ihrer Gesamtheit sind sie aber schon sehr mächtig.

Neben den Inputs, Outputs, Parametern und Internals, die an verschiedenen Stellen genutzt werden können, stehen zwei Register zur Verfügung: V0 und V1. Werden mehr Zwischenwerte benötigt, müssen Internals genutzt werden.

Die Befehle werden in einem Dokument, das mit dem BrainDesigner heruntergeladen werden kann, detailliert beschrieben. In diesem Dokument finden sich auch Hinweise, wie diese Befehle in der ARM-Architektur umgesetzt werden können. ARM-Prozessoren werden in vielen eingebetteten Systemen und somit auch Robotern verwendet.

Plugins/Player-Settings

Alle Eingangs- und Ausgangswerte (sei es aus Dateien oder von an den PC angeschlossenen Robotern oder Geräten) und alle Berechnungen werden von Plugins durchgeführt. Diese sind zum Teil direkt in den BrainDesigner integriert, zum Teil liegen sie in Form von DLL-Dateien im BrainDesigner-Unterordner `plugins`.

Jedes Plugin kann verschiedene Funktionen bereitstellen. Bei In-/Outputs sind dies im Normalfall jeweils eine Funktion zum Liefern von Eingangswerten und eine zum Schreiben von Ausgabewerten. Bei Berechnungs-Plugins ist dies meist nur eine Funktion „Calculate“.

In jedem Zeitschritt werden die Funktionen verschiedener Plugins in einer bestimmten zeitlichen Abfolge aufgerufen. Normalerweise werden zunächst alle Inputs gelesen, dann alle Berechnungen durchgeführt und dann alle Outputs geschrieben. Eine solche Abfolge wird als „Configuration“ bezeichnet. Im „Settings“-Dialog im Menü unter „Tools“ -> „Settings...“ können Configurations geändert oder hinzugefügt werden. Standardmäßig gibt es eine Configuration „Standard“. Die aktive Configuration wird über eine DropDown-Box ausgewählt. Mit dem grünen Plus-Zeichen rechts davon kann eine neue Configuration hinzugefügt werden, mit dem roten X wird eine Configuration gelöscht.

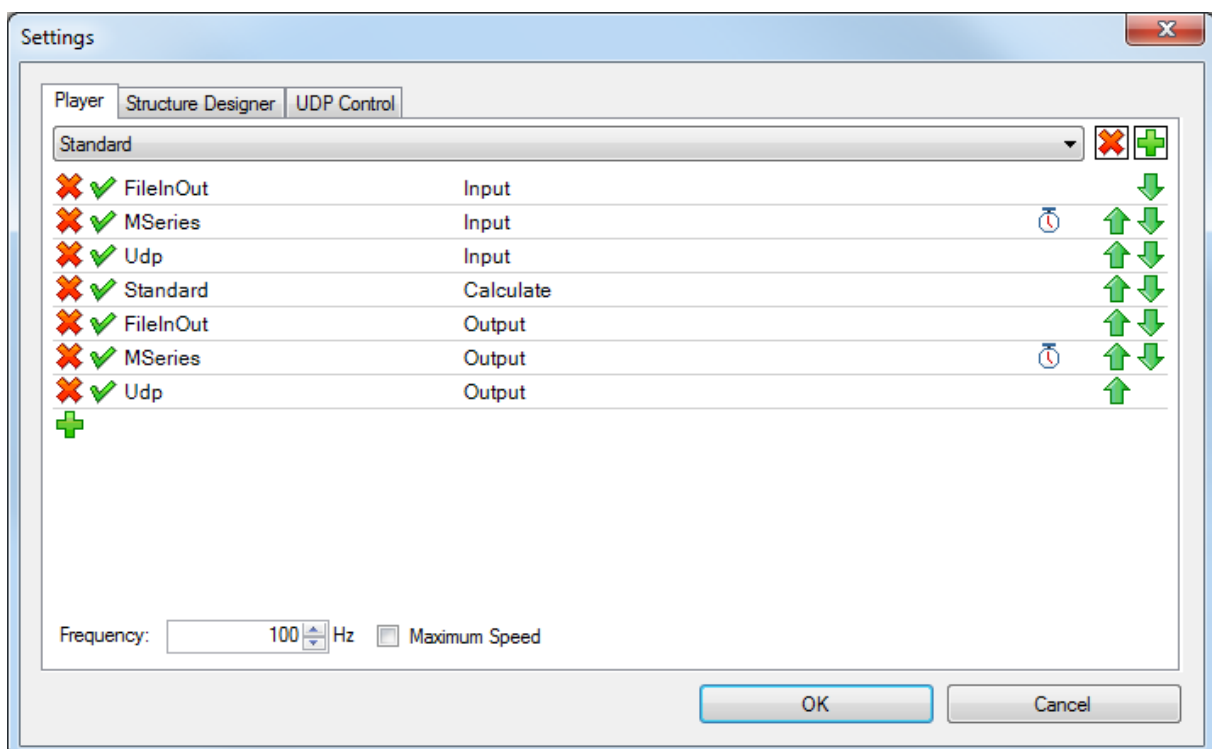


Abbildung 3: Der Settings-Dialog mit einer Standard-Konfiguration

Unterhalb der Configuration-Auswahl sieht man alle aktivierten Funktionen. Mit dem roten X kann eine Funktion gelöscht werden, mit dem grünen Haken wird eine Funktion deaktiviert. Über die grünen Pfeile am Ende jeder Funktionszeile kann die Reihenfolge geändert werden. Vor den grünen Pfeilen kann je nach Plugin ein Zahnrad und/oder ein Uhr-Symbol angezeigt werden. Das Uhr-Symbol zeigt an, dass der Zeittakt, mit dem das neuronale Netz berechnet wird, von diesem Plugin vorgege-

ben wird. Ist mindestens ein Plugin aktiv, bei dem das Uhr-Symbol angezeigt wird, wird die Einstellung der Berechnungsgeschwindigkeit am unteren Rand des Dialogs ignoriert. Wird ein Zahnrad-Symbol angezeigt, kann auf dieses geklickt werden, um das Plugin zu konfigurieren.

Der erste Eintrag jeder Zeile zeigt das Plugin an (dies kann per Klick geändert werden). Dahinter wird die gewählte Funktion angezeigt und kann ebenfalls per Klick geändert werden.

Eine übliche Configuration ist beispielsweise zunächst die Funktion „Input“ vom Plugin „FileInOut“, dann die Funktion „Calculate“ vom Plugin „Standard“ und dann die Funktion „Output“ vom Plugin „FileInOut“.

Plugins, die nicht verwendet werden (z.B. wenn keine Datei gelesen oder geschrieben wird), müssen normalerweise nicht deaktiviert werden. Verschiedene Configurations sind nur nötig, um unterschiedliche – sich gegenseitig ausschließende – Plugin-Konfigurationen zu betreiben. Beispielsweise kann man eine Configuration haben, bei der alles auf dem PC berechnet wird und nur die Ein- und Ausgänge mit einem Roboter verbunden sind und eine Configuration, bei dem das gleiche Netz auf den Roboter gespielt werden soll, damit dieses dort autonom berechnet wird. Dann muss im Settings-Dialog die Configuration entsprechend umgeschaltet werden.

Der BrainDesigner kommt mit Plugins für verschiedene Anwendungen. Neben dem Standard-Plugin für Berechnungen am PC gibt es Inputs und Outputs für Dateien („FileInOut“ für Textdateien; „SoundInOut“ für Wave-Dateien) und für Roboter („A-Series“, „M-Series“, „Semni“). Das Plugin „MSeriesDeployment“ übersetzt ByteCode für den Roboter Myon (M-Serie) des Labors für Neurorobotik in ausführbaren ARM-Code und spielt diesen auf den Roboter. Mit dem Plugin „Udp“ können beliebige andere Programme mit dem BrainDesigner kommunizieren.