

VERHALTENSSTEUERUNG EINER UNIVERSELLEN KOGNITIVEN BENUTZERSCHNITTSTELLE ANHAND EINER HEIZUNGSSTEUERUNG

Werner Meyer*, Markus Huber**, Matthias Wolff*

*Brandenburgische Technische Universität Cottbus-Senftenberg, **InnoTec21 GmbH Leipzig
werner.meyer@b-tu.de

Kurzfassung: Die in diesem Paper vorgestellte Verhaltenssteuerung wurde im UCUI-Projekt (Universal Cognitive User Interface) entwickelt [1]. Das Ziel dieses Projekts ist die Hardware-Realisierung einer autonomen kognitiven Nutzerschnittstelle mit konstruktionsbedingter strikter Wahrung der Privatsphäre. Das Systemverhalten wird mittels Merkmal-Werte-Relationen (MWRs) realisiert. Dabei werden zwei unterschiedliche Paradigmen eingesetzt. Zum einem kann eine instinktive Weiterverarbeitung der Eingabe-MWR erfolgen und zum anderen besteht die Möglichkeit, dass das System auf Basis der bisher gelernten Nutzerinteraktionen Entscheidungen trifft. Die Nutzerinteraktionen werden mit Hilfe eines Markow-Entscheidungsprozesses gelernt. Um die Größe des Zustandsraumes zu beschränken, wird er variabel gehalten. Für die Aktion und Perzeption werden Transduktoren eingesetzt, die die Schnittstelle zwischen MWRs und der Spracheingabe bzw. Sprachausgabe darstellen. [2]

1 Einleitung

Im Informationszeitalter nehmen Daten einen wichtigen Stellenwert ein. Vor allem die Wirtschaft profitiert durch das Sammeln von Daten. Der Begriff Big-Data beschreibt dabei eine schierere Menge an Daten, die per Hand nicht mehr ausgewertet werden kann, weshalb die Auswertung maschinell erfolgen muss. Die so gewonnenen Informationen werden profitbringend eingesetzt oder an mögliche Interessenten verkauft. Die Befürchtung eines möglichen Missbrauchs der gesammelten Daten rückt gegenwärtig immer wieder in den Vordergrund. Das Interesse an Datenschutz nimmt vermehrt zu, weshalb alternative Technologien benötigt werden, die das gleiche Potential besitzen wie die herkömmliche Technik.

Das Projekt *Universal Cognitive User Interface* (UCUI) soll dabei zeigen, dass die Anwendung einer kognitiven Benutzerschnittstelle auch unter Einhaltung der Privatsphäre möglich ist.

1.1 Universal Cognitive User Interface

Im Projekt UCUI wird das Ziel verfolgt, eine Benutzerschnittstelle zu entwickeln, die sich auf den Anwender einstellt und dennoch dessen Privatsphäre schützt. Durch die Anpassung an den Anwender, soll beispielsweise die Steuerung von Haushaltsgeräten auch von weniger technisch affinen Personen ermöglicht werden. Als Beispielanwendung dient eine Heizungssteuerung. Der Anwender hat die Möglichkeit, einzeln oder auch gemischt eine Sprach-, Gestik- oder Bildschirmeingabe zu tätigen. Die Bearbeitung der eingegebenen Daten erfolgt ausschließlich lokal, weshalb das System keine Internetanbindung benötigt. Lediglich der Benutzer besitzt die Entscheidungsgewalt eine Cloud-Verbindung zu ermöglichen. Aktuelle Systeme können nur durch Anwendung von Big-Data eine (meist) korrekte Interpretation der Anwendereingabe vornehmen. Um bei UCUI ein adäquates Systemverhalten ohne Anwendung von Big-Data zu realisieren, werden semantische Strukturen verwendet.

1.2 Merkmal-Werte-Relation

„Die Interpretation von sprachlichem Verhalten und die Artikulation einer Bedeutungsstruktur bildet [sic] die Schnittstelle zwischen der syntaktischen und der semantischen Ebene eines hierarchischen Sprachdialogsystems.“ [3, S. 52] Innerhalb der semantischen Ebene werden Merkmal-Werte-Relationen (MWRs) [4] eingesetzt, welche die Bedeutung der Ein- und Ausgabe repräsentieren. Bei einer MWR handelt es sich um eine Baumstruktur, deren Knoten semantische Anker repräsentieren und deren Blätter Werte enthalten können. Werden semantische Anker oder Werte durch einen Platzhalter offen gelassen, so handelt es sich um ein semantisches Schema [4]. In Tabelle 1 sind die für unser System relevanten Merkmale der semantischen Anker mit deren Bedeutung aufgelistet. In dem hier präsentierten System basiert die Verhaltenssteuerung auf Berechnungen mit MWRs und das Weltmodell wird durch semantische Schemata definiert (siehe Unterabschnitt 2.1). Damit der Verhaltenssteuerung eine MWR übergeben werden kann, muss diese aus der Eingabe des Anwenders erzeugt werden. Unter Anwendung der Spracheingabe soll hier ein Beispiel demonstriert werden. Für die Über-

Tabelle 1 – Semantische Anker und deren Bedeutung

sem. Anker	Bedeutung
FVR	Beginn der MWR
PROG	Programm (Heizungssteuerung)
STEP	Auszuführender Schritt
EXE	Ausführung einer Anweisung
REQUST	Nachfrage
INFOEXE	Information über Anweisung
MODE	Modus des System (an/aus)
HEATABS	Absoluter Temperatur
TEMPTARGET	Soll-Heizwert
TEMPIS	Ist-Temperaturwert
DATETIME	Zeitangabe
TIMEABS	Uhrzeit
HOURLDIM	Stundenangabe
MINUTEDIM	Minutenangabe
DATEABS	Datumsangabe
MONTH	Monat
DAY	Tag ...
OFWEEK	... der Woche
OFMONTH	... des Monats

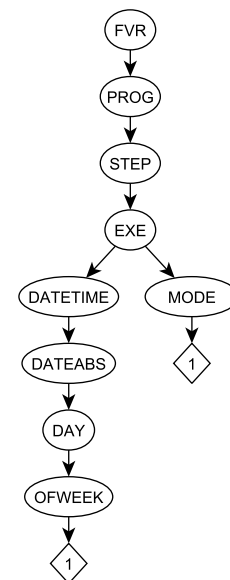


Abbildung 1 – MWR zur Aussage „Schalte die Heizung am Montag an.“

setzung der Spracheingabe wird ein Äußerungs-Bedeutungs-Transduktor verwendet, der im System die Grammatik repräsentiert [4, 2]. Als Eingabesymbole dienen die Wörter der vom Anwender getätigten Äußerung. Ausgabesymbole sind die entsprechenden Merkmale und Werte. Mit der Aussage „Schalte die Heizung am Montag an.“ wird die in Abbildung 1 dargestellte MWR erzeugt. Das Merkmal *PROG* steht für ein entsprechendes Programm. In der Demonstration beziehen wir uns ausschließlich auf die Heizungssteuerung, daher wird an dieser Position (vorerst) kein anderes Merkmal genannt. Das Merkmal *STEP* bezieht sich auf die auszuführenden Schritte. Durch eine kaskadierte Anordnung [4] dieses Merkmals kann eine MWR mehrere Anweisungen aus einer Äußerung repräsentieren. Da (ebenfalls vorerst) nur einfache Anweisungen berücksichtigt werden, entfällt eine kaskadierte Anordnung „welche eine syntaktische

Reihenfolge kodieren“ [5] in unserer Demonstration. Die ersten drei Merkmale unterscheiden sich in den hier vorgestellten MWRs nicht, daher werden diese in den nachfolgenden Abbildungen nicht weiter dargestellt. Für die Ausgabe einer MWR wird ein Bedeutungs-Äußerungs-Transduktor als Grammatik verwendet, der aus der MWR eine Wortfolge generiert, die per Sprachsynthese ausgegeben wird.

2 Instinktive Verhaltenssteuerung

Die Verhaltenssteuerung bildet den zentralen Punkt zwischen äußerem und innerem Kreis des kognitiven Systems und ist hier auf die Arbeit mit MWRs ausgelegt. Ein Flussdiagramm der Verhaltenssteuerung ist in Abbildung 2 dargestellt. Die Verarbeitung der Eingabe-MWR erfolgt in den drei Schritten: Vereinigung, Abgleich und Erwartungsgenerierung. In den folgenden Abschnitten werden die einzelnen Arbeitsschritte anhand einer Beispieleingabe erläutert. Eine Vereinigung tritt dann auf, wenn das System bereits eine Erwartungs-MWR durch einen laufenden Dialog bereithält. Ist keine Erwartung vorhanden, so wird die Eingabe direkt zum Abgleich weitergegeben. Daher findet sich der Abschnitt zur Vereinigung hier an letzter Stelle.

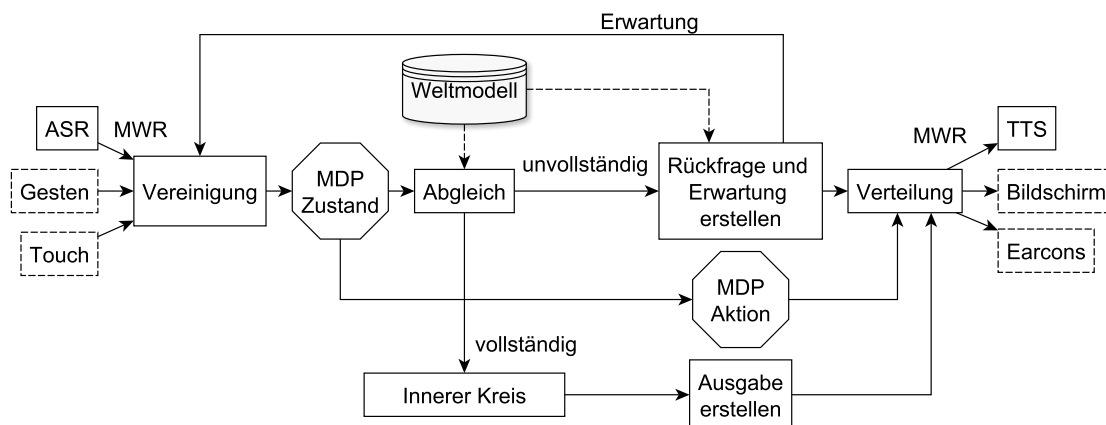


Abbildung 2 – UCUI - Verhaltenssteuerung

2.1 Abgleich

Beim Abgleich ermittelt das System ein passendes Schema aus dem Weltmodell. Im Weltmodell sind alle Schemata hinterlegt, die eine für das System sinnvolle Bearbeitung der eingegebenen Informationen ermöglichen. So ist zum Beispiel bei der Eingabe des Einschaltens, neben dem Merkmal *MODE* mit dem Wert 1 auch die Information der Temperatureinstellung mit dem Merkmal *TEMPTARGET* und einem Platzhalter für einen beliebigen Temperaturwert vorhanden (siehe Abbildung 3 links). Besitzt das Merkmal *MODE* den Wert 0 ist die Information der Temperatureinstellung nicht relevant und das benachbarte Merkmal *TEMPTARGET* nicht vorhanden (siehe Abbildung 3 rechts). Insgesamt besitzt das Weltmodell der Heizungssteuerung 15 Schemata. Um das passende Schema zu ermitteln, wird die Anzahl der übereinstimmenden Merkmale im Verhältnis zum Schema gezählt. Werte werden nur dann berücksichtigt, wenn diese im Schema vorhanden sind. Ist dies der Fall, so muss der gleiche Wert in der Eingabe-MWR ebenfalls vorhanden sein. Dabei erhält ein Schema mit einem übereinstimmenden Wert eine höhere Priorität als alle übrigen Schemata. Bei der für die Beispieleingabe getätigten Äußerung „Schalte die Heizung am Montag an.“ (siehe Abbildung 1) wurde ein Schema ermittelt, bei dem die Werte für die Merkmale *TEMPTARGET* der Temperatur, *DATETIME* der Uhrzeit und *DAY* → *OFWEEK* für den Wochentag durch Platzhalter offengelassen sind (siehe Abbil-

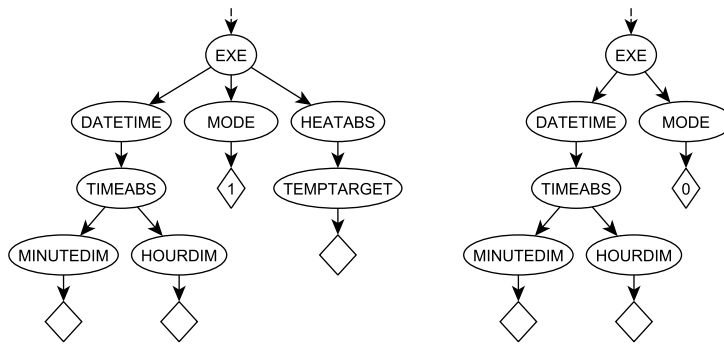


Abbildung 3 – Zwei von 15 Schemata aus dem Weltmodell der Heizungssteuerung

dung 3 links). Nach der Wahl eines Schemas werden alle Merkmale und Platzhalter für Werte, der Eingabe zugeordnet. Eine MWR gilt dabei als vollständig, wenn alle Platzhalter durch Werte ersetzt wurden. Ist eine MWR vollständig, so wird diese zur Bearbeitung an den inneren kognitiven Kreis gegeben. Sind noch Platzhalter vorhanden, so wird die MWR als Erwartung für die nächste Nutzereingabe weitergegeben.

2.2 Erwartung und Rückfrage

Eine Erwartungs-MWR wird generiert, wenn das System mit dem Anwender in einem laufenden Dialog steht. Dies ist dann der Fall, wenn die Eingabe nicht ausreicht um eine Anweisung auszuführen. Das System stellt Rückfragen und hält die bereits vorhandenen Informationen in der Erwartungs-MWR vor. Aus der vom Abgleich übermittelten MWR werden die Erwartung und die Rückfrage generiert (siehe Abbildung 4). Die Erwartung besteht aus allen gegebenen Merkmalen und vorhandenen Werten. Die Generierung der Rückfrage erfolgt aus allen Pfaden, deren Blätter mit einem Platzhalter endet. Da vorerst eine einfache Grammatik für die Sprachsynthese berücksichtigt wird und das System auch in der Lage sein muss, einfache Fragen zu äußern, erfolgt die zufällige Wahl eines dieser Pfade. Der Platzhalter wird nicht übernommen. Das Merkmal *EXE* wird mit dem Merkmal *REQUEST* für die Darstellung einer Frage des Systems an den Anwender ausgetauscht.

Die Erwartung wird an den Eingang des Systems zur Vereinigung mit der nächsten Eingabe gesendet. Die Rückfrage (hier im Beispiel nach der Temperatureinstellung) wird per Sprachsynthese dem Anwender übermittelt.

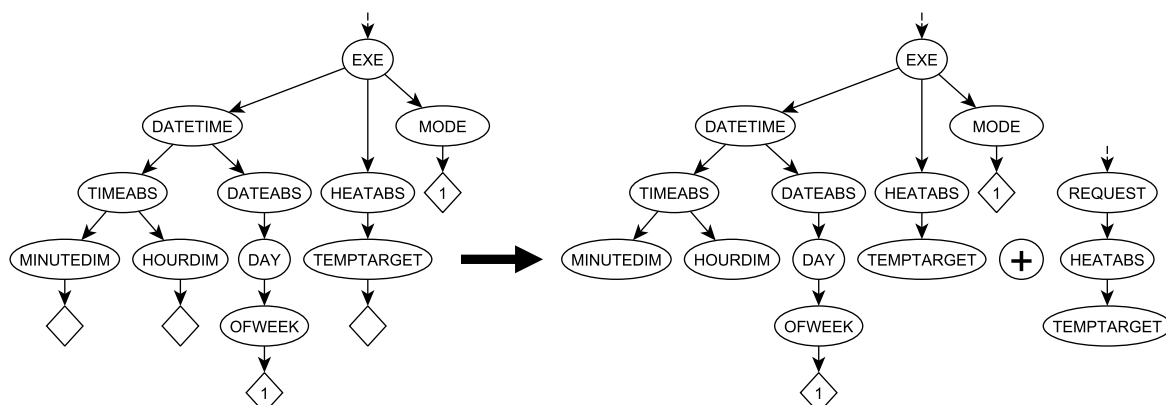


Abbildung 4 – v.l.n.r. MWR vom Abgleich → Erwartungs-MWR + Rückfrage

2.3 Vereinigung

Eine Vereinigung findet statt, wenn nach einer Eingabe eine Erwartung im System bereit gehalten wird. Nehmen wir an, der Nutzer antwortet auf die Rückfrage nach der einzustellen-

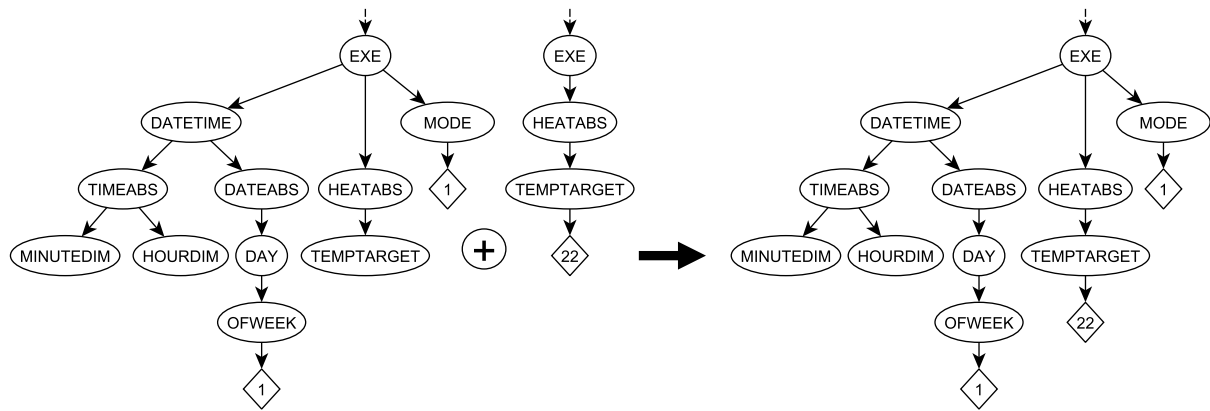


Abbildung 5 – Vereinigung zweier MWRs. V.l.n.r. Erwartung + Eingabe → Vereinigung.

den Temperatur mit "22 Grad". Die entsprechende Eingabe-MWR ist in Abbildung 5 (Mitte) dargestellt. Sie wird nun mit der Erwartung vereinigt. Das Resultat der Vereinigung ist die in Abbildung 5 (rechts) angegebene MWR. Das System berücksichtigt bei der Vereinigung, dass jedes Merkmal der Eingabe-MWR bereits in der Erwartungs-MWR vorhanden sein muss (siehe Abbildung 5). Ist dies nicht der Fall, entspricht die Eingabe nicht der Erwartung. Die Erwartung wird verworfen, der vorherige Dialog abgebrochen und die Eingabe unbearbeitet an den Abgleich weitergeleitet. Tritt der Fall ein, dass unter dem selben Merkmal in Eingabe und Erwartung ein Wert auftritt, gibt es vier Strategien, die bei der Erstellung der Vereinigung berücksichtigt werden müssen.

- Wert aus der Eingabe-MWR übernehmen → Korrektur des Anwenders, die vom System nicht weiter berücksichtigt wird.
- Wert aus der Erwartungs-MWR beibehalten → Strikte Ausführung einer einmal ermittelten Werteingabe.
- Beide Werte behalten → Spätere Entscheidung unter Unsicherheit treffen.
- Beide Werte verwerfen → Generiert erneut Rückfragen des Systems.

In der aktuellen Version des UCUI-Geräts kommt die letzte Variante zum Einsatz. Eine Verfeinerung ist für zukünftige Versionen geplant.

3 Adaptive Verhaltenssteuerung

Neben dem instinktiven Verhalten kann das System auch auf bereits gelernte Nutzereingaben zurückgreifen, um angemessene Aktionen auszuführen. Dabei wird das Q-Lernen in einem dynamisch generierten Zustandsraum angewendet.

3.1 Strategielernen

Ein Zustand wird im System durch eine unvollständige oder vollständige MWR (einige bzw. alle Werte vorhanden) beschrieben, wie sie im Ergebnis der Vereinigung entsteht (siehe Abbildung 2). Die potenzielle Größe des Zustandsraumes entspricht demnach der Anzahl möglicher Wertekombinationen pro Schema, im Beispielschema also Uhrzeit (24 Stunden und 60 Minuten), Wochentag (7), Monatstag (31), Monat (12), Temperatureingabe (0 – 40°C in 1-Grad-Schritten) und dem Modus des Systems (An/Aus). Berücksichtigt werden muss, dass einige Werte nicht gemeinsam auftreten. Zum Beispiel, ist bei einem Ausschaltvorgang des Systems

die Temperatureinstellung nicht relevant oder bei der Nennung des Wochentages wird kein Montag erwähnt. Die Größe des Zustandsraumes ergibt sich daher wie folgt:

$$24 \cdot 60 \cdot (31 \cdot 12 + 7) \cdot (40 + 1) = 22.376.160 \quad (1)$$

Zu beachten ist, dass die Größe des Zustandsraumes lediglich die Möglichkeit der Einstellungen des Systems angibt, jedoch nicht die Notwendigkeit, dass jede Konstellation der Einstellung auch benötigt wird. Es ist beispielsweise nicht zu erwarten, dass ein Anwender nach der Einstellung einer Temperatur für 17 Uhr, eine erneute Einstellung der Temperatur für 17:01 Uhr vornehmen wird. Also müssten bei dieser Vielzahl nicht alle Zustände im System berücksichtigt werden. Ein solches Nutzerverhalten kann vom System dennoch nicht ausgeschlossen werden.

Relevant sind alle Zustände, von denen das System über den Nutzer lernen kann. Daher werden die Zustände direkt nach der Vereinigung von Eingabe- und Erwartungs-MWR (siehe Abbildung 2) generiert. Aus jeder hier auftretenden MWR wird ein – der MWR entsprechender – Zustand generiert und die dazugehörige MWR gespeichert. Sollte die gleiche MWR zu einem späteren Zeitpunkt erneut auftreten, so wird der bereits generierte Zustand verwendet. Durch einen auftretenden Dialog zwischen Anwender und instinktiver Verhaltenssteuerung entstehen weitere Zustände, deren MWRs zunehmend mehr Werte enthalten als in den vorherigen Zuständen. Erhält die instinktive Verhaltenssteuerung eine vollständige MWR, so wird dieser Zustand als ein finaler Zustand festgelegt. Nach dem Erreichen eines finalen Zustandes werden rückwirkend über die Transitionen der besuchten Zustände Belohnungswerte nach der Q-Funktion (2) des Markow-Entscheidungsprozess verteilt [6].

$$Q(z, a) = r(z) + \gamma \cdot \max_a Q(z', a') \quad (2)$$

Die Belohnung $r(z)$ wird vergeben, wenn z ein finaler Zustand ist. Der Faktor $\max_a Q(z', a')$ entspricht der Nachfolgetransition mit der größten Belohnung. Der Diskontierungsfaktor γ ist für eine Schwächung der Belohnungswerte des aktuellen Zustands gegenüber des Nachfolgezustands zuständig und muss einen Wert von $0 < \gamma < 1$ einnehmen.

Für die instinktive Verhaltenssteuerung kann jede Transition eine andere Aktion darstellen. Aus der Perspektive der adaptiven Verhaltenssteuerung gibt es nur zwei Aktionen. Zum einen das Weiterverfolgen der Strategie π aus Gleichung (3), die einen Pfad im Zustandsraum ermittelt und zum anderen das Senden der im Zustand hinterlegten MWR an den Abgleich der instinktiven Verhaltenssteuerung. Lediglich im finalen Zustand muss die hinterlegte MWR an den inneren kognitiven Kreis übergeben werden. Die vollständige MWR wird ausgeführt und der Anwender über die getätigte Aktion informiert.

Wird der im Beispiel erwähnte Dialog wiederholt, so werden die bereits vorhandenen Transitionen gestärkt. Überschreiten die Belohnungswerte einen Schwellwert, wird die instinktive Verhaltenssteuerung übersprungen und die Transition nach der Strategie π in Gleichung (3) verwendet.

$$\pi = r(z) + \gamma \cdot \arg \max_a Q(z', a') \quad (3)$$

Durch das wiederholte Auftreten von gleichen Dialogen lernt das System, die gewünschten Einstellungen des Anwenders zu verwenden. Bei der Beispieleingabe bedeutet dies, dass das System auf die erneute Aussage „Schalte die Heizung am Montag an“ ohne erneute Rückfragen die gelernten Einstellungen vornimmt und dem Anwender mit der Äußerung „Ich stelle die Heizung am Montag um 17 Uhr auf 22°C ein“ dies signalisiert.

3.2 Vergessen und Korrektur

Die Größe des Zustandsraums wird durch zwei Parameter begrenzt: Zum einem durch die bereits erwähnte Konstellation an Einstellungsmöglichkeiten und zum anderen durch den verfüg-

baren Speicher. Sollen noch weitere Parameter, wie die Nutzeridentifikation oder die Steuerung von weiteren Haushaltsgeräten in Abhängigkeiten von Sensoren im System berücksichtigt werden, ist zu beachten, dass der Speicher nicht ausreichen könnte. Daher ist es notwendig, dass Zustände auch gelöscht werden können.

Das Löschen der Zustände kann nicht beliebig erfolgen, da die Relevanz der Zustände unterschiedlich ausfällt. Die ältesten und zuerst angelegten Zustände können zugleich die meist genutzten Zustände im System sein. Das ausschließliche Löschen der jüngsten Zustände führt dagegen zu einer transienten globalen Amnesie, was einer krankhaften Störung des Kurzzeitgedächtnisses entspräche. Daher ist es notwendig, dass eine Strategie Anwendung findet, die alle vorhandenen Zustände berücksichtigt. Unter der Beachtung von drei Punkten, wird der Zustand mit der geringsten Bewertung gelöscht:

- Das Löschen während eines laufenden Dialoges ist verboten, da erst nach Erreichen eines finalen Zustands die Belohnungen im Zustandsraum angepasst werden.
- Die zuletzt erstellten Zustände dürfen nicht gelöscht werden, da sonst die letzten Eingaben des Anwenders verworfen werden und das System nicht mehr in der Lage ist, neue Strategien zu lernen.
- Die Bewertungen aller Transitionen im Zustandsraum müssen herabgesetzt werden, um auch das Löschen älterer Zustände – die nach einer Zeit für den Anwender an Bedeutung verloren haben – zu ermöglichen.

Der letzte Punkt wird durch Anwendung der Gleichung (4) realisiert.

$$Q(z, a) = \gamma \cdot Q(z, a) \quad (4)$$

Damit der Anwender die Möglichkeit besitzt auf die Äußerung „Schalte die Heizung am Montag an“ mit einer anderen, als dem vom System hinterlegten Bedeutung zu entgegnen, müssen auch Korrekturen zugelassen werden. Eine Korrektur wird vorgenommen, wenn nach der Äußerung des Systems der Anwender innerhalb eines Zeitrahmens eine erneute Anweisung tätigt. Die Korrektur ähnelt dabei dem Prinzip des Löschens. Die Belohnungen der Transitionen des verwendeten Pfads im Zustandsraum werden nach Gleichung (4) herabgesetzt.

4 Zusammenfassung und Ausblick

Im Zuge des UCUI-Projektes kann die hier vorgestellte Verhaltenssteuerung einen ersten adäquaten Ansatz zu den gegenwärtigen digitalen Sprachassistenten bieten. Neben stereotypem „instinktiven“ Verhalten ist das implementierte System auch in der Lage, sich auf das Nutzerverhalten einzustellen. Dazu gehört auch ein kontrolliertes Vergessen.

Gegenwärtig ist das Projekt noch nicht abgeschlossen. Ziel ist es, Grammatik und Weltmodell zu erweitern sowie die Strategien zum Vergessen und zur Korrektur zu verfeinern. Durch das Einbringen weiterer Parameter besitzt das System das Potential eine umfangreiche Benutzerschnittstelle zu bieten.

Danksagung

Die vorliegende Arbeit wurde im Rahmen des BMBF-Verbundprojektes „Entwicklung eines universellen Hardware-Moduls für eine intuitive Interaktion mit beliebigen Geräten“ unter dem Kennzeichen 16SV7304 gefördert.

Literatur

- [1] DUCKHORN, F., M. HUBER, W. MEYER, O. JOKISCH, C. TSCHÖPE, und M. WOLFF: *Towards an Autarkic Embedded Cognitive User Interface*. In *Interspeech2017*, S. 3435f. 2017.
- [2] LINDEMANN, J.: *Semantische Interpretation und Artikulation mit Äußerungs-Bedeutungs-Transduktoren*. In *27. Konferenz Elektronische Sprachsignalverarbeitung (ESSV)*, S. 119–126. Jokisch, O., Leipzig, 2016.
- [3] LINDEMANN, J.: *Interpretation und Artikulation mit Äußerungs-Bedeutungs-Transduktoren*. Dissertation, Cottbus, 2016.
- [4] WIRSCHING, G. und M. WOLFF: *Semantische Dekodierung von Sprachsignalen am Beispiel einer Mikrofonfeldsteuerung*. In *25. Konferenz Elektronische Sprachsignalverarbeitung*, S. 104–109. Dresden, 2014.
- [5] WIRSCHING, G. und M. WOLFF: *Semantische Dekodierung von Sprachsignalen am Beispiel einer Mikrofonfeldsteuerung*. In R. HOFFMANN (Hrsg.), *Elektronische Sprachsignalverarbeitung 2014, Tagungsband der 25. Konferenz Dresden, 26. 28. März 2014*, S. 104 – 109. 2014.
- [6] WOLFGANG, E.: . In 2 (Hrsg.), *Grundkurs Künstliche Intelligenz*. Vieweg+Teubner Verlag, 2009.