14. Workshop Fuzzy-Systems und Computational Intelligence, Dortmund, 2004; ISBN 3-937300-20-1, ISSN 1614-5267

Temporal erweiterte Prädikate der Fuzzy-Logik zur Überwachung und Wartung

Thorsten W. Schmidt, Dominik Henrich
Universität Bayreuth
Fakultät für Mathematik, Physik und Informatik
Lehrstuhl Angewandte Informatik III
D-95440 Bayreuth
E-Mail: {thorsten.w.schmidt, dominik.henrich}@uni-bayreuth.de
Http://ai3.inf.uni-bayreuth.de

1. Einleitung

1.1. Motivation

Seit vielen Jahren schon beschäftigt sich die Forschung mit Fuzzy-Logik und ihrer Anwendung in Fuzzy-Reglern Seit einigen Jahren gibt es auch schon industriell eingesetzte Fuzzy-Regler. Diese werden zum Beispiel in Waschmaschinen oder anderen Geräten des häuslichen Gebrauchs verwendet. Die Vorteile der Fuzzy-Logik sind, dass vorhandenes Wissen über ein Regelungsprozess sehr leicht zur Modellierung eines Reglers verwendet werden kann und im weiteren Verlauf der Entwicklung und Verbesserung des Reglers dieses Wissen immer transparent bleibt und somit nicht verloren geht. Durch die klare Lesbarkeit der Fuzzy-Logik bleibt das Wissen auch wartbar. Diese Fuzzy-Regler können aber nicht als Wartungssystem eingesetzt werden, da sie nicht in der Lage sind zeitliche Abhängigkeiten von Ereignissen untereinander oder überhaupt Zeit zu modellieren, wie dies beim Modell Checking möglich ist, welches Temporale Logik verwendet [Karjoth87]. Aus diesem Grund ist eine Erweiterung der Fuzzy-Logik um zeitliche Aspekte nötig. Es existieren ein paar Ansätze, um die Zeit in Fuzzy-Logik einzubringen. In Kapitel 2 werden vier solcher Ansätze vorgestellt.

1.2. Problemstellung

Ein Problem bei Fuzzy-Reglern ist, dass es keine sachlich motivierten Ansätze gibt, durch welche klar ersichtlich wird, wie zeitliche Aspekte in Fuzzy-Logik gehandhabt werden können. Model Checking mit Temporaler Logik ist eine in sich geschlossene Methodik. Die temporalen Prädikate können zur Beschreibung von Szenarien genutzt werden um Prozesszustände mit zeitlichen Abhängigkeiten zu modellieren. Das Problem, welches im folgenden gelöst wird, beschäftigt sich nun damit, Prädikate der Temporalen Logik in die Fuzzy-Logik zu übersetzen und zwar so, dass zum Einen die so genannte Temporale Fuzzy-Logik genauso mächtig ist wie die Temporale Logik und zum Anderen eine solide mathematische Basis geschaffen wird, so dass die Bedingungen, welche an Fuzzy-Prädikate und Zugehörigkeitsfunktionen gestellt werden auch erfüllt sind. Gesucht ist demnach die Vereinigung der Temporalen Logik und Fuzzy-Logik zur *Temporalen Fuzzy-Logik*.

Im weiteren Verlauf dieses Artikels beschreibt das zweite Kapitel den Stand der Forschung während das dritte Kapitel die verwendeten Abkürzungen und Definitionen für die späteren Kapitel beschreibt. Kapitel 4 zeigt die Einreihung der Temporalen Fuzzy-Logik im Vergleich zur temporalen/atemporalen beziehungsweise scharfen/unscharfen Logik. Das nächste, fünfte Kapitel definiert alle Prädikate der Temporalen Fuzzy-Logik. Kapitel 6 beschreibt eine Anwendung der Temporalen Fuzzy-Logik an einem konkreten Beispiel. Kapitel 7 fasst die Arbeit zusammen und beschreibt die daraus gewonnenen Schlussfolgerungen.

1.3. Abgrenzung

Ein *Überwachungssystem* ist ein von einem zu überwachenden Prozess unabhängiges System, welches diesen Prozess in seinem Verhalten mittels Sensoren überwacht. Prozesskennzahlen informieren dabei über die internen, nicht zwingenderweise bekannten, Zustände des Prozesses. Bei den Aktuatoren, ist nicht immer bekannt, welchen quantitativen Einfluss diese auf den Prozess haben, somit ist eine Steuerung des Prozesses nicht möglich. Ist jedoch der qualitative Einfluss bekannt, so kann der Prozess geregelt werden. Werden nun in dem Verhalten des Prozesses Abweichungen zu den gewünschten benutzerdefinierten Vorgaben erkannt, kann ein *Regler* in das Verhalten des Prozesses eingreifen und Parameter so verändern, dass das Verhalten des Prozesses sich dem Verhalten nähert, welches von einem Benutzer gewünscht wird [Castillo02]. Ein einfaches Beispiel für ein Überwachungssystem ist die Überwachung der Helligkeit in einem Raum. Sinkt die Helligkeit unter einen angegebenen Schwellwert, so erkennt dies das Überwachungssystem und gibt dem Prozess den Auftrag den Raum stärker zu beleuchten, indem mehr Lampen angeschaltet werden.

Ein *vorausschauendes Überwachungssystem* benutzt nicht nur aktuelle Sensordaten aus dem Prozess, sondern auch mögliche Sensordaten aus der Zukunft [Fantoni00], [Palit00]. Natürlich ist es nicht möglich diese zukünftigen Daten zu messen. Sie müssen mit geeigneten Methoden aus dem bekannten vergangenen Signalverlauf vorhergesagt werden. Werden diese zukünftigen Sensorwerte an ein Überwachungssystem gegeben, so kann dieses eine zukünftige Abweichung vom gewünschten, benutzerdefinierten Verhalten feststellen. Das Eintreten der Abweichung ist dabei nicht garantiert; sie muss nicht eintreten. Im Gegenteil, dadurch, dass dem Überwachungssystem bekannt ist, was bei den aktuellen Parameterwerten in der Zukunft passieren würde, können schon frühzeitig Maßnahmen ergriffen werden, um ein anderes Verhalten herbei zu führen. Wenn nun im obigen Beispiel Lampen verwendet werden, welche eine lange Zeit benötigen, um ihre maximale Helligkeit zu erreichen (Neonröhren mit einem sehr hohen Wirkungsgrad: ca. 15 Minuten, Energiesparlampen: ca. 10 Minuten oder ähnliche), so genügt ein nicht vorausschauendes Überwachungssystem nicht mehr. Das vorausschauende Überwachungssystem kann jedoch feststellen, dass es im Raum immer dunkler wird. Bevor es im Raum zu dunkel ist, also die Helligkeit den angegebenen Schwellwert unterschreitet, schaltet das Überwachungssystem weitere Lampen ein.

Ein *Wartungssystem* baut meistens, nicht immer, auf einem Diagnosesystem auf [Althoff92]. Das in dieser Arbeit vorzustellende Wartungssystem baut jedoch auf einem (vorausschauenden) Überwachungssystem auf. Dann, wenn ein Überwachungssystem durch Veränderung der Prozessparameter keine Verbesserung mehr erreichen kann und sich die Prozesskennzahlen nicht innerhalb eines tolerierbaren Bereiches befinden, liegt ein *Fehler* im System vor, welcher nicht ausgeglichen werden kann. Dieser Fehler kann zum Beispiel eine defekte Teilkomponente sein, welche ersetzt werden muss. Das Wartungssystem generiert in diesem Fall einen Wartungsauftrag für einen Benutzer und teilt diesem mit, welche Teilkomponente einen Fehler verursacht haben könnte. Durch den vorausschauenden Aspekt eines Wartungssystems können Ausfälle dieser Art frühzeitig vorhergesagt und hierfür Wartungsaufträge generiert werden. Die Wartungsaufträge können zeitlich in der Zukunft datiert sein, da der vorhergesagte Ausfall nicht unmittelbar, sondern in der Zukunft eintritt. Das System wird so lange betrieben, wie es funktionsfähig bleibt, also seine Prozesskennzahlen in einem tolerierbaren Bereich liegen. Die Abstände zwischen verschiedenen Wartungen, bei welchen ein

Bediener die Maschine anhält und sie repariert sind maximal. Dadurch, dass die Wartungsaufträge auch in der Zukunft liegen können und es so möglich ist mehrere Wartungsaufträge zu sammeln und zu einem Zeitpunkt alle Wartungsarbeiten parallel auszuführen, wird der Prozess nur einmal angehalten und somit die Standzeiten (*Wartungszeiten*) verringert. So entsteht kein außerplanmäßiger Produktionsausfall, denn die Wartungsarbeiten können eventuell zu Zeiten geringer Auslastung durchgeführt werden. Im obigen Beispiel entspricht dies dem Überwachungssystem, welches versucht den Raum durch Einschalten weiterer Lampen zu erhellen. Da aber defekte Lampen existieren, liegt die Helligkeit auch dann noch unter dem gegebenen Schwellwert, wenn alle Lampen angeschaltet sind. In diesem Fall generiert das Wartungssystem einen Wartungsauftrag, in welchem es dem Benutzer mitteilt, defekte Lampen im Raum auszutauschen. Eine andere Möglichkeit, dass ein Wartungssystem einen Wartungsauftrag generiert, ist wenn festgestellt wird, dass Lampen Anzeichen für einen baldigen Defekt aufweisen, so dass dann nicht mehr genügend Licht produziert werden kann.

Als Endziel, welches über die Temporalen Fuzzy-Prädikate dieses Papers hinaus geht, möchten wir ein Wartungssystem, wie oben beschrieben entwickeln. Das Wartungssystem generiert Wartungsaufgaben für einen zu wartenden Prozess. Die Wartungsaufgaben werden dabei durch Fuzzy-Logik Regeln beschrieben. Natürlich kann das Wartungssystem auch als Überwachungssystem oder Regelungssystem verwendet werden. Das Hauptaugenmerk liegt jedoch auf dem Wartungssystem, welches Wartungsaufträge generiert. Die automatisch generierten Wartungsaufträge müssen dabei zeitlich so geplant sein, dass eine möglichst ökonomische Abarbeitung ohne größere Standzeiten des Prozesses möglich ist.

2. Stand der Forschung

In der Forschungsabteilung der Firma Flender Service GmbH werden spezielle Sensoren zur Überwachung entwickelt [Flender01], [Flender02]. Diese Sensoren dienen der Überwachung von beispielsweise Fräsmaschinen, welche Lager, Zahnräder und andere Komponenten mit hohem Verschleiß beinhalten. Zur Beobachtung der Komponenten mit hohem Verschleiß werden Schwingungssensoren verwendet, denn die Erfahrung hat gezeigt, dass sich das Schwingungsprofil dieser Komponenten charakteristisch mit dem Erreichen des Endes der Lebensdauer verändert. Das heißt die Schwingungen werden aufgenommen, zur Steuereinheit übertragen und dort Fouriertransformiert. Aus den Signalverläufen werden unter Verwendung von mit diesen Maschinen gesammelten Wissen Rückschlüsse auf die weitere Lebensdauer der Komponenten gezogen, so dass diese ausgetauscht werden können bevor es zu einem Ausfall kommt. Einsetzbar ist dieses System in allen Maschinen, an welchen über Jahre hinweg der Verschleiß dieser Komponenten gemessen und protokolliert wurde. Die Einschränkung bei diesem Ansatz liegt dabei, Einzelteile einer Maschine zu beobachten, welche sich bewegen beziehungsweise hohen mechanischen Kräften ausgesetzt sind und durch diese Bewegung verschleißen beziehungsweise durch die Kräfte ermüden. Dies Stellt den Überwachungsanteil des Systems dar. Wartungsarbeiten werden vorgenommen, wenn das Überwachungssystem eine Wartungsfirma über das Internet über einen baldigen Ausfall informiert. Dies ist ein Wartungssystem, welches gezielt auf eine Aufgabe zugeschnitten ist. Unser Wartungssystem soll aber allgemeiner und flexibel einsetzbar sein. Außerdem ist unser Ansatz intuitiver und näher an der menschlichen Ausdrucksweise. Durch Verwendung von Fuzzy-Logik ist er mathematisch formalisiert und dadurch nachvollziehbar. Im Folgenden werden vier Ansätze vorgestellt, welche den Begriff der Zeit in Fuzzy-Logik einbringen.

Unter anderem in [Fick00] werden Takagi Sugeno Regeln [Takagi85] in einem Fuzzy-Regler zeitlich abhängig gemacht, indem die Zeit *t* als weitere Eingabevariable in den Bedingungsteil der Regeln mit aufgenommen wird. Zum Beispiel „**IF** *expression* **AND** $t_0 - t$ **IS** *now* **THEN** *action*“. Dies hat zur Folge, dass Regeln nur zu gewissen Zeiten feuern oder wie bei [Fick00] neu dargestellt, jede Bedingung einer Regel mit einem Zeitintervall konjunktiv verknüpft werden muss. Die Regelaktivierung wird um so höher sein, je näher der aktuelle Zeitpunkt am gegebenen Zeitpunkt liegt. Solch ein Konstrukt ist dann nützlich, wenn Regeln nur zu einer bestimmten Tageszeit feuern dürfen. Zum Beispiel „$(t_0 - t)$ modulo Tag **IS** *Mittagszeit*“. In diesem Ansatz ist die Zeit zwar unscharf, aber es gibt keine Möglichkeit, Ereignisse untereinander auf zeitliche Abhängigkeiten zu vergleichen, da nur eine Zeitangabe pro Regel vorgesehen ist. Diese Vergleichbarkeit wird jedoch von uns für ein Wartungssystem gefordert.

In [Bovenkamp97] wird ein neuer Ansatz zum temporalen Schließen mit Fuzzy-Logik vorgestellt. So genannte Fuzzy-Zeit-Objekte werden definiert, um Unschärfe in Fakten und Zeit zu repräsentieren. Fuzzy-Zeit-Objekte sind zweistellige, einwertige Zugehörigkeitsfunktionen der Form $\mu_{f \times z}(x,t) := min\left(\mu_f(x), \mu_z(t)\right)$. Für diese Zugehörigkeitsfunktionen muss für den Fakt $\max_x \mu_f(x) = 1$ und die Zeit $\max_t \mu_z(t) = 1$ gelten. Damit gilt auch die Separierbarkeit $\forall x \forall t \in \{t | \max_t \mu_z(t) = 1\} : \mu_f(x) \overset{!}{=} \mu_{f \times z}(x,t)$ beziehungsweise $\forall t \forall x \in \{x | \max_x \mu_f(x) = 1\} : \mu_z(t) \overset{!}{=} \mu_{f \times z}(x,t)$. Man kann also ein Fuzzy-Zeit-Objekt genau dann wenn die Separierbarkeit gilt in Fakt und Zeit zerlegen, so dass man die ursprünglichen Fuzzy Terme μ_f(x) beziehungsweise $\mu_z(t)$ wieder erhält, ohne dass die Zeit die Unschärfe des Fakts beeinflusst. Die Beziehung zwischen Fakt und Zeit wird durch das temporales Schließen gebildet. Auch bei zeitlich oder faktisch eingeschränkten Fuzzy-Zeit-Objekten gilt dieser Zusammenhang, wobei Fuzzy-Zeit-Objekte nicht gemischt werden dürfen. Die Fuzzy-Zeit-Objekte sind nicht intuitiv verwendbar und erlauben keine UND beziehungsweise ODER Verknüpfungen von Fuzzy-Zeit-Objekten, aber es ist möglich ein Fuzzy-Zeit-Objekt in einer Konklusion zu verwenden. Dieser Ansatz reicht noch nicht ganz um unsere geforderte Mächtigkeit, beliebige Verwendung von zeitlichen Abhängigkeiten in Bedingung und Konklusion einer Fuzzy-Regel, zu erfüllen.

In [Lamine01] wird die Lineare Temporale Logik (LTL) genutzt um zeitliche Abhängigkeiten in Programme einzubringen. Die Syntax der LTL ist ähnlich zu dem der Programmiersprache C. Die LTL ist unscharf in der Auswertung von Bedingungen, denn es wird mit Wahrscheinlichkeiten beziehungsweise Wahrscheinlichkeitsverteilungen für ihr Eintreten gerechnet. Aber es werden keine natürlichsprachlichen Konstrukte verwendet (auch keine Fuzzy-Inferenz). Somit steht die LTL also zwischen der Temporalen Logik und der Fuzzy-Logik. Die Zeit in einer Bedingung ist durch das Prädikat **always** mit dem Zeitintervall [0,?] gegeben. Das angegebene Zeitintervall steht für „von Jetzt (0) bis in alle Ewigkeit (?)“. Das vorgestellte Anwendungsgebiet, ist ein Roboter, welcher auf einem Straßennetz fährt, wobei die Wahrscheinlichkeitsverteilungen für den weiteren Straßenverlauf bekannt sind. Die einzelnen Regeln werden dann dazu genutzt, um in einer Regelschleife die Programmteile zu aktivieren, welche bei den aktuell vorliegenden und vermuteten zukünftigen Bedingungen mit hoher Wahrscheinlichkeit das gewünschte Ergebnis liefern. Es sind UND beziehungsweise ODER Verknüpfungen von Zeitintervallen möglich, aber es wird hierzu keine Fuzzy-Logik beziehungsweise eine Inferenz darauf verwendet, da die Zeitintervalle nur darüber entscheiden, ob eine eine Regel überhaupt in Betracht gezogen wird.

In [Cárdenas02] wird die Fuzzy Temporal Constraint Logic (FTCL), eine Erweiterung von Prolog mit Horn-Klauseln um Fuzzy-Logik und temporale Prädikate, präsentiert. Die temporalen Prädikate **before**, **after** und **at_the_same_time** ermöglichen es, zeitliche Abhängigkeiten von Ereignissen untereinander auszudrücken. Die Zeit ist dabei als unscharf anzusehen. Außerdem sollen alle Ereignisse, welche zu einer gegebenen Zeit eintreten, nur mit Ereignissen im gleichen Zeitintervall verglichen werden, da Relationen zu unterschiedlichen Zeiten nicht möglich sind. Ein Beispiel, welches zeigt, dass der in [Cárdenas02] vorgestellte Ansatz nicht für ein Wartungssystem verwendet werden kann, ist ein Ereignis, welches kausal von einem anderen Ereignis abhängt, wie zum Beispiel: „Gestern war es heiß und heute leckt das Rohr, dann existiert heute ein Folgeschaden durch die Hitzeeinwirkung“.

Wie an den gezeigten Beispielen zu sehen ist, gehen die verwandten Arbeiten in die von uns angestrebte Richtung, Zeit beziehungsweise zeitliche Abhängigkeiten von Bedingungen in Fuzzy-Regler einzubinden. Jedoch gibt es noch keine Lösung, welche dies anhand von der schon ausgereiften Temporalen Logik gemacht hat. Auch wurde noch kein Fuzzy-Regler zur Wartung verwendet. Dies ist durch [Giron02] ersichtlich, wo mehr als 200 Fuzzy-Regler in mehr als 20 Kategorien klassifiziert werden. Ein Fuzzy-Regler mit Temporaler Fuzzy-Logik oder ein Fuzzy-Regler als Wartungssystem existieren demnach noch nicht. Diese Entwicklung ist unser Ziel, welches wir schrittweise erreichen möchten. Ein Schritt für diese Entwicklung ist die hier vorgestellte Erweiterung der Fuzzy-Logik zur Temporalen Fuzzy-Logik.

3. Abkürzungen und Definitionen

t_A Zeitpunkt in der Vergangenheit, bis zu welchem Aufzeichnungen von Sensordaten eines Prozesses vorhanden sind. Sensordaten vor diesem Zeitpunkt sind nicht bekannt.

t_O Zeitpunkt in der Zukunft, bis zu welchem Aufzeichnungen von Sensordaten eines Prozesses vorhanden sind, oder bis zu welchem Zeitpunkt eine Vorhersage noch Sinn macht, denn Vorhersagen werden immer ungenauer, je weiter in die Zukunft geschaut wird.

t_C Aktueller Zeitpunkt eines Prozesses $t_C \in\ ]t_A, t_O[$. Es existiert immer ein Teilintervall mit Sensordaten des Prozesses für die Vergangenheit $[t_A, t_C]$ und für die Zukunft $]t_C, t_O]$.

t_i Ein beliebiger Zeitpunkt aus dem Intervall $t_i \in [t_A, t_O]$, welcher wenn nicht anders genannt sowohl in der Zukunft, als auch in der Vergangenheit beziehungsweise Gegenwart liegen kann.

T_i Zeitintervall von t_C bis t_i, wobei t_C größer oder kleiner als t_i sein kann. T_i ist demnach definiert als: $T_i := \begin{cases} [t_C, t_i] & , t_C < t_i \\ [t_i, t_C] & , t_i > t_C \end{cases}$.

$h^i(t)$ Datum h, welches vom Sensor S^i zum Zeitpunkt $t \leq t_C$ aufgezeichnet wurde.

$p^i(t)$ Datum p, welches für den Sensor S^i für den Zeitpunkt $t \geq t_C$ vorhergesagt wird.

$S^i(t)$ Sensorwert des i-ten Sensors zum Zeitpunkt t des Prozesses. Liegt der Zeitpunkt $t > t_C$ in der Zukunft, so wird eine Vorhersage zurückgeliefert. Liegt er jedoch in der Vergangenheit $t < t_C$ oder Gegenwart $t = t_C$, dann wird ein aufgezeichneter

Wert zurückgeliefert. $S^i(t)$ ist definiert als:

$$S^i(t) := \begin{cases} h^i(t) & , t \leq t_C \\ p^i(t) & , t > t_C \end{cases}.$$

A^i Aktuator i des Prozesses, durch welchen der Regler mit seinen Ausgaben Einfluss auf den Prozess nimmt.

ft Fuzzy-Term $ft = \{(x_i, y_i) | x_i < x_j, 0 \leq y_i \leq 1, i < j, 0 \leq i, j < n\}$, welcher durch die Punkte (x_i, y_i) eines Polygonzugs der Länge n dargestellt wird. In den meisten Fällen wird $n = 3$ und $y_0 = y_{n-1} = 0$ gelten und so, mit dem Fuzzy-Term, ein gleichschenkeliges Dreieck dargestellt. Da ein Fuzzy-Term beliebig viele Stützstellen n besitzen kann, können beliebige stetige Funktionen durch lineare Interpolation approximiert werden.

$\mu_{ft}(x)$ Ist eine Zugehörigkeitsfunktion für den Fuzzy-Term ft, für welche die Bedingungen nach [Bothe95] gelten:

- $\forall x \in \mathbb{R} : \mu_{ft}(x) \geq 0$
- $\mu_{ft}(x)$ umso größer, je besser x ein Bewertungskriterium eines Experten erfüllt
- Einheitsintervall-Normalisierung, so dass für $\mu(x)$: $x \to [0,1]$
- Normalisierung auf ein Bezugselement x_0 mit $\mu_{ft}(x_0) = 1$

Während die ersten beiden Bedingungen harte Bedingungen sind, welche in jedem Fall erfüllt sein müssen, sind die dritte und vierte Bedingung weiche Bedingungen, welche nicht unbedingt erfüllt sein müssen. Sind sie nicht erfüllt, so sprechen wir von einem unpräzisem, ungenauem oder auch unsicherem Fuzzy-Term, ansonsten von einem präzisen, genauen, oder sicheren Fuzzy-Term. Wenn nicht anders genannt, ist mit einem Fuzzy-Term immer ein präziser Fuzzy-Term gemeint. Die Einheitsintervall-Normalisierung impliziert nicht notwendigerweise die Normalisierung auf ein Bezugselement, denn die Zugehörigkeitsfunktionen müssen keine surjektiven Abbildungen sein.

Zugehörigkeitsfunktionen sind hier stetige, stückweise lineare, nicht notwendigerweise monotone Funktionen deren Stützstellen durch den Fuzzy-Term ft gegeben sind. Sie sind im gesamten Definitionsbereich definiert und es gilt folgende Definition:

$$\mu_{ft}(x) = \begin{cases} y_i + (y_{i+1} - y_i) \cdot \dfrac{(x - x_i)}{(x_{i+1} - x_i)} & , x_i \leq x < x_{i+1} \\ y_0 & , x < x_0 \\ y_{n-1} & , x \geq x_{n-1} \end{cases}$$

Alternativ kann auch eine höherdimensionale Interpolation der Stützstellen verwendet werden. Wir beschränken uns aber bewusst auf stückweise lineare Funktionen, denn mit diesen berechnet sich die Vereinigung von Fuzzy-Termen, zum Beispiel $ft_A \cup ft_B$, und die Akkumulierung mit der Schwerpunktsmethode nach [Watanabe86] (Berechnung von Integralen) sehr schnell und effizient und führt so zu effizienteren und schneller berechenbaren Fuzzy-Reglern.

Man spricht bei $\mu_{ft}(x)$ von der Zugehörigkeit von x zum Fuzzy-Term ft oder von der Aktivierung des Fuzzy-Terms ft durch x.

4. Temporale Fuzzy-Logik

Betrachtet man die Logik mit der Unterteilung, wie in Tabelle 1 dargestellt, in atemporale und temporale Logik auf der Zeitachse beziehungsweise in scharfe und unscharfe Logik auf der Schärfeachse, so repräsentiert die Prädikatenlogik, welche den Wertebereich {0, 1} besitzt und zeitlich konstant ist, die Gruppe der scharfen, atemporalen Logik. Die Prädikatenlogik kann nun zum Einen zeitlich erweitert werden oder zum Anderen unscharf gemacht werden.

Die zeitliche Erweiterung ist für die Modellierung von dynamischen Systemen wie Zustandsautomaten, physikalischen Prozessen und ähnlichem nötig. Also erweitert man die Prädikatenlogik um Operatoren, welche es erlauben zeitliche Abhängigkeiten zu beschreiben. Diese temporalen Operatoren beschreibt [Karjoth87] für Zustandsautomaten beziehungsweise für Prozesse mit zeitlicher Diskretisierung. Zu den Operatoren werden Kalküle und Herleitungen (Beweise), welche in der Prädikatenlogik gültig sind, so angepasst, dass sie auch in der Temporalen Logik gültig sind. Somit ist die Prädikatenlogik ein Spezialfall der Temporalen Logik ohne Verwendung der temporalen Operatoren. Die temporalen Operatoren sind unten links in Tabelle 1 erläutert.

Die unscharfe Erweiterung der Prädikatenlogik kommt Anwendungsgebieten mit ungenauem oder unscharfem Wissen zu gute. Denn bei diesen Anwendungen existiert Wissen von einem oder mehreren Experten, welches in der Regel unscharf formuliert ist. Da die Fuzzy-Logik dazu gedacht ist unscharfes Wissen darzustellen und zu verarbeiten, kann das unscharfe Expertenwissen einfacher in Fuzzy-Logik als in Prädikatenlogik ausgedrückt werden. Die Fuzzy-Logik erweitert wie die Temporale Logik ebenfalls die Prädikatenlogik, führt aber keine neuen Operatoren ein, sondern weicht den Wertebereich von {0, 1} beziehungsweise {falsch, wahr} zum Intervall [0,1] auf. Wobei $x = 0$ falsch und $x = 1$ wahr entspricht. Dass dadurch ebenfalls alle Beweise und Kalküle aus der Prädikatenlogik gelten wurde schon mehrfach in der Literatur gezeigt [Bothe95], [Karjoth87]. Die Prädikatenlogik ist demnach auch ein Spezialfall der Fuzzy-Logik.

Um nun die Temporale Fuzzy-Logik, welche einerseits unscharf und andererseits temporal ist, zu erhalten gibt es zwei Möglichkeiten. Entweder wird der Wertebereich der Temporalen Logik unscharf gemacht, oder die Fuzzy-Logik erhält wie die Temporale Logik temporale Operatoren, um zeitliche Abhängigkeiten zu beschreiben. Vergleicht man in Tabelle 1 die Temporale Logik mit der Temporalen Fuzzy-Logik, so sieht man, dass für jeden temporalen Operator ein Fuzzy-Prädikat existiert. Ganz offensichtlich ist die Temporale Fuzzy-Logik (TFL) eine Mischung der Temporalen Logik (TL) und Fuzzy-Logik (FL), da von beiden die Erweiterungen zur Prädikaten Logik (PL) eingeflossen sind. So erhält man durch weglassen der Unschärfe von der TFL die TL und durch weglassen der temporalen Prädikate von der TFL die FL.

Die Darstellung in der Tabelle lässt durch die Prädikate **WILL_EXIST_NEXT** und **PREEXIST_PREVIOUS** den Schluss zu, dass die Temporale Fuzzy-Logik wie die Temporale Logik nur für zeitlich diskrete Prozesse geeignet ist. Dass dem nicht so ist, wird im nächsten Abschnitt erläutert. Dort wird gezeigt, wie die unscharfen temporalen Prädikate nach dem menschlichen Empfinden intuitiv modelliert sind und dass sie auch zum Beschreiben für zeitlich kontinuierliche Prozesse eingesetzt werden können.

	Scharf		Unscharf	
Atemporal	Prädikatenlogik (PL)		Fuzzy-Logik (FL)	
	$P(x) \rightarrow \{0,1\}$		$P(x, ft) \rightarrow [0,1]$	
	$P(x)$: X ist wahr		x **IS** ft: x gehört zu μ_{ft}	
Temporal	Temporale Logik (TL)		Temporale Fuzzy-Logik (TFL)	
	$P(x) \rightarrow \{0,1\}$ $P(x, y) \rightarrow \{0,1\}$		Einfach: $P(x, ft, t_i, t_j) \rightarrow [0,1]$ Komplex: $P(x, y, ft_A, ft_B) \rightarrow [0,1]$	
	$\Box x$	x wird immer wahr sein	x **WILL_BE** ft	x wird immer zu μ_{ft} gehören $\forall t > t_C: \mu_{ft}(S^i(t)) > 0$
	$\boxminus x$	x war immer wahr	x **WAS** ft	x gehörte immer zu μ_{ft} $\forall t < t_C: \mu_{ft}(S^i(t)) > 0$
	$\Diamond x$	x wird mindestens einmal wahr sein	x **WILL_EXIST** ft	x wird min. einmal zu μ_{ft} gehören $\exists t > t_C: \mu_{ft}(S^i(t)) > 0$
	$\diamondminus x$	x war mindestens einmal wahr	x **PREEXIST** ft	x gehörte min. einmal zu μ_{ft} $\exists t < t_C: \mu_{ft}(S^i(t)) > 0$
	$\bigcirc x$	x wird zum nächsten Zeitpunkt wahr	x **WILL_EXIST_NEXT** ft	x gehört im nächsten Zeitpunkt zu μ_{ft} $\mu_{ft}(S^i(t+\Delta t)) > 0$
	Θx	x war zum letzten Zeitpunkt wahr	x **PREEXIST_PREV.** ft	x gehörte im letzten Zeitpunkt zu μ_{ft} $\mu_{ft}(S^i(t-\Delta t)) > 0$
	$x\,\mathcal{U}\,y$	x bleibt wahr bis y wahr wird	x **IS** ft_A **UNTIL** y **IS** ft_B	x gehört zu μ_{ft} bis y zu μ_{ft} gehört
	$x\,\mathcal{S}\,y$	x gilt seit y zum letzten mal wahr geworden ist	x **IS** ft_A **SINCE** y **IS** ft_B	x gehört zu μ_{ftA} seit y mal zu μ_{ftB} gehörte

Tabelle 1: Einordnung der Temporalen Fuzzy-Logik in Bezug zur scharfen/unscharfen beziehungsweise atemporalen/temporalen Logik.

5. Temporale Fuzzy-Prädikate

Prädikate bestimmen den Zugehörigkeitsgrad von Sensordaten zu gegebenen Fuzzy-Termen. Die Anzahl der Sensoren beziehungsweise Sensordaten, welche von einem Prädikat betrachtet werden ist n. *Einfache* temporale Prädikate (**WILL_BE**, **WAS**, ...) besitzen einen Fuzzy-Term und sind (n+3)-stellige Abbildungen $P(\vec{x}, ft, t_i, t_j) \rightarrow [0,1]$, wenn ein Zeitintervall gegeben ist beziehungsweise (n+1)-stellige Abbildungen $P(\vec{x}, ft) \rightarrow [0,1]$, wenn kein Zeitintervall gegeben ist. *Komplexe* temporale Prädikate (**SINCE**, **UNTIL**) besitzen zwei Fuzzy-Terme und sind (2n+2)-stellig Abbildungen $P(\vec{x}, \vec{y}, ft_A, ft_B) \rightarrow [0,1]$. Für alle Prädikate müssen folgende Bedingungen erfüllt sein:

- Normierung: $\forall \vec{x}, ft: 0 \leq P(\vec{x}, ft) \leq 1$ (1)
- Stetigkeit: für beliebige, aber feste *ft* gilt:
  $\forall \vec{x_0} \forall \epsilon > 0 \;\; \exists |\vec{\delta}| > 0: \forall \vec{x}\, mit |\vec{x} - \vec{x_0}| < |\vec{\delta}|\, folgt |P(\vec{x}, ft) - P(\vec{x_0}, ft)| < \epsilon$ (2)
- Komplement (Standardnegation): ¬P ist definiert durch: $\neg P(\vec{x}, ft) = 1 - P(\vec{x}, ft)$ (3)

Definition: Prädikat **IS**

$$S^i \,\mathbf{IS}\, ft := P_\nu(i, t_C, ft) := \mu_{ft}(S^i(t_C))$$

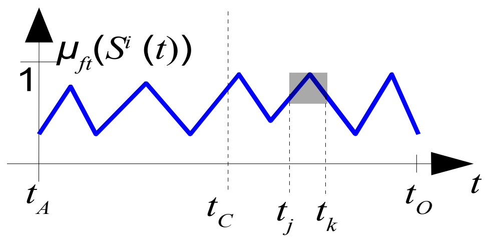
Erläuterung: Die Prädikatenfunktion P_ν steht für *now* und das Prädikat **IS** entspricht dem schon bekannten Prädikat **IS** der Fuzzy-Logik. Die Semantik ist, dass je genauer der zuletzt aufgezeichnete, also aktuelle Sensorwert S^i (t_C) des i-ten Sensors im gegebenen Fuzzy-Term *ft* liegt, desto höher ist dessen Aktivierungsgrad, welcher durch die

Zugehörigkeitsfunktion μ_{ft} bestimmt ist. Durch die Einschränkung auf präzise Fuzzy-Terme (siehe Definition von μ_{ft} in Kapitel 3) ist die Zugehörigkeitsfunktion und somit auch das Prädikat normiert (1). Die Zugehörigkeitsfunktion liefert einen Wert zwischen 0 (Datum aktiviert nicht den Fuzzy-Term) und 1 (Datum aktiviert den Fuzzy-Term) und entspricht dem Aktivierungsgrad des Fuzzy-Terms *ft*. Der Aktivierungsgrad entspricht in diesem Fall auch dem Wahrheitsgehalt des Prädikates. Des weiteren ist das Prädikat eine stetige Funktion, denn die Zugehörigkeitsfunktion ist als stückweise lineare und somit stetige Funktion definiert (2). Für die Negation des Prädikates gilt (3):

$$\begin{aligned}
&\neg P_\nu(i, t_C, ft) = \neg\mu_{ft}(S^i(t_C)) = \mu_{\neg ft}(S^i(t_C)) \\
&= \mu_{\neg\{(x_i, y_i) | x_i \le x_j, 0 \le y_i \le 1, i<j, 0 \le i, j<n\}}(S^i(t_C)) \\
&= \mu_{\{(x_i, 1-y_i) | x_i \le x_j, 0 \le y_i \le 1, i<j, 0 \le i, j<n\}}(S^i(t_C)) \\
&= 1 - \mu_{ft}(S^i(t_C)) = 1 - P_\nu(i, t_C, ft)
\end{aligned}$$

<u>Definition: Prädikat</u> <u>**WILL_BE**</u>

$$S^i \text{ } \mathbf{WILL\_BE}[t_j, t_k] ft := P_\Phi(S^i, t_j, t_k, ft) := \frac{1}{t_k - t_j} \int_{t_j}^{t_k} P_\nu(i, t, ft)\, dt$$

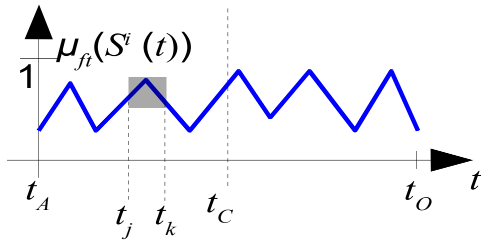


<u>Erläuterung:</u> Die Prädikatenfunktion P_Φ steht für *future-whole*. Das Prädikat **WILL_BE** berechnet die Aktivierung des Fuzzy-Terms *ft* für Sensordaten aus dem Zeitintervall $[t_j, t_k]$. **WILL_BE** ist umso aktiver, je genauer die vorhergesagten Sensordaten $p^i(t)$ mit $t \in [t_j, t_k]$ im gegebenen Fuzzy-Term *ft* liegen. Streifen die Sensordaten den Fuzzy-Term nur kurz, so ist die Aktivierung geringer, als wenn die Sensordaten während der gesamten Intervalldauer im Fuzzy-Term liegen würden. Das gegebene Zeitintervall liegt dabei vollständig in der Zukunft, demnach gilt $t_j > t_C$. Die Aktivierung wird durch das Mittel über alle Sensorwerte im gegebenen Zeitintervall mit dem Prädikat **IS** gebildet. Da P_ν normiert ist, also immer kleiner gleich 1 und größer 0 ist, ist das Integral maximal $t_k - t_j$. Das Integral wird durch $t_k - t_j$ geteilt und ist so normiert auf das Intervall [0,1] (1). Das Prädikat P_Φ ist stetig, denn P_ν ist auch stetig und das Integral einer stetigen Funktion ist wiederum stetig. Der konstante Vorfaktor $1/(t_k - t_j)$ hat auch keinen Einfluss auf die Stetigkeit (2). Für die Negation des Prädikates gilt (3):

$$\begin{aligned}
&\neg P_\Phi(S^i, t_j, t_k, ft) = \frac{1}{t_k - t_j} \int_{t_j}^{t_k} \neg P_\nu(i, t, ft)\, dt \\
&= \frac{1}{t_k - t_j} \int_{t_j}^{t_k} 1 - P_\nu(i, t, ft)\, dt = 1 - \frac{1}{t_k - t_j} \int_{t_j}^{t_k} P_\nu(i, t, ft)\, dt \\
&= 1 - P_\Phi(S^i, t_j, t_k, ft)
\end{aligned}$$

<u>Definition: Prädikat</u> <u>**WAS**</u>

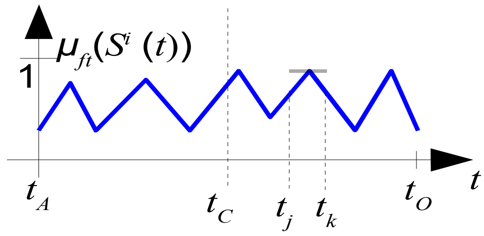
$$S^i \text{ } \mathbf{WAS}[t_j, t_k] ft := P_\Pi(S^i, t_j, t_k, ft) := \frac{1}{t_k - t_j} \int_{t_j}^{t_k} P_\nu(i, t, ft)\, dt$$



Erläuterung: Die Prädikatenfunktion P_Π steht für *past-whole*. Das Prädikat **WAS** berechnet die Aktivierung des Fuzzy-Terms *ft* für Sensordaten aus dem Zeitintervall $[t_j,\ t_k]$. **WAS** ist umso aktiver, je genauer die vergangenen Sensordaten $h^i\ (t)$ mit $t \in [t_j,\ t_k]$ im gegebenen Fuzzy-Term *ft* liegen. Streifen die Sensordaten das Intervall nur kurz, so ist die Aktivierung geringer, als wenn die Sensordaten während der gesamten Intervalldauer im Fuzzy-Term liegen würden. Das gegebene Zeitintervall liegt dabei vollständig in der Vergangenheit, demnach gilt $t_k < t_C$. Die Aktivierung wird durch das Mittel über alle Sensorwerte im gegebenen Zeitintervall mit dem Prädikat **IS** gebildet. Die Bedingungen (1) – (3) gelten für das Prädikat P_Π analog zu P_Φ mit den selben Begründungen.

Definition: Prädikat **WILL_EXIST**

$$S^i\ \textbf{WILL\_EXIST}[t_j,\ t_k]\ ft := P_\varphi(S^i, t_j, t_k, ft) := \max_{t\in[t_k, t_k]} P_\nu(i, t, ft)$$

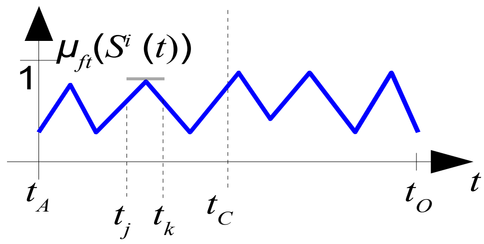


Erläuterung: Die Prädikatenfunktion P_φ steht für *future-single*. Das Prädikat **WILL_EXIST** bewertet, ähnlich wie das Prädikat **WILL_BE**, die Aktivierung des Fuzzy-Terms *ft* für Sensordaten aus dem Zeitintervall $[t_j,\ t_k]$. **WILL_EXIST** prüft jedoch, ob überhaupt ein vorhergesagter Sensorwert $p^i\ (t)$ mit $t \in [t_j,\ t_k]$ im gegebenen Fuzzy-Term *ft* liegt. Dabei ist es für die Aktivierung des Fuzzy-Terms egal, ob ein oder mehrere Sensorwerte im Fuzzy-Term *ft* liegen. Für die Aktivierung zählt lediglich der Sensorwert mit der maximalen Aktivierung. Die Aktivierung wird über die Maximumsuche aller Aktivierungen von Sensorwerten im gegebenen Zeitintervall mit dem Prädikat **IS** gebildet. Da P_ν normiert ist, ist auch das Maximum von P_ν normiert (1). Die Komposition von stetigen Funktionen mit der Maximumfunktion liefert immer stetige Funktionen, und da P_ν stetig ist, ist somit auch P_φ stetig (2). Für die Negation des Prädikates gilt (3):

$$\begin{aligned}
&\neg P_\varphi(S^i, t_j, t_k, ft) = \max_{t\in[t_k, t_k]} \neg P_\nu(i, t, ft)\\
&= \max_{t\in[t_k, t_k]} 1 - P_\nu(i, t, ft) = 1 - \max_{t\in[t_k, t_k]} P_\nu(i, t, ft)\\
&= 1 - P_\varphi(S^i, t_j, t_k, ft)
\end{aligned}$$

Definition: Prädikat **PREEXIST**

$$S^i\ \textbf{PREEXIST}[t_j,\ t_k]\ ft := P_\pi(S^i, t_j, t_k, ft) := \max_{t\in[t_k, t_k]} P_\nu(i, t, ft)$$



Erläuterung: Die Prädikatenfunktion P_π steht für *past-single*. Das Prädikat **PREEXIST** bewertet, ähnlich wie das Prädikat **WAS**, die Aktivierung des Fuzzy-Terms *ft* für Sensordaten aus dem Zeitintervall $[t_j,\ t_k]$. **PREEXIST** jedoch prüft, ob überhaupt ein vergangener Sensorwert $h^i\ (t)$ mit $t \in [t_j, t_k]$ im gegebenen Fuzzy-Term *ft* liegt. Dabei ist es für die Aktivierung des Fuzzy-Terms egal, ob ein oder mehrere Sensorwerte im Fuzzy-Term *ft* liegen. Für die Aktivierung zählt lediglich der Sensorwert mit der maximalen Aktivierung. Die Aktivierung wird über die Maximumsuche aller Aktivierungen von Sensorwerten im gegebenen Zeitintervall mit dem Prädikat **IS** gebildet. Die Bedingungen (1) – (3) gelten für das Prädikat P_π analog zu P_φ mit den selben Begründungen.

Definition: Prädikat **WILL_EXIST_NEXT**

$$S^i \textbf{ WILL\_EXIST\_NEXT } ft \equiv S^i \textbf{ WILL\_EXIST}[t_C, t_C+\Delta t]\, ft \equiv P_\varphi(S^i, t_C, t_C+\Delta t, ft)$$

Erläuterung: Das Prädikat **WILL_EXIST_NEXT** ist ein Makro für das Prädikat **WILL_EXIST** und entstammt aus der direkten Übersetzung des temporalen Operators Ox (wird zum nächsten Zeitpunkt beziehungsweise im nächsten Zustand wahr, siehe Tabelle 1, Kapitel 4) und kann in der Temporalen Fuzzy-Logik mit kontinuierlicher und diskreter Zeit bei einer geeigneten Wahl von Δt verwendet werden. Bei der Auswertung des Prädikates interessiert nur ein kleiner Verlauf Δt in der Zukunft. Es ist zu beachten, dass Δt so gewählt werden muss, dass immer $\Delta t \leq t_O - t_C$ gilt. Bei Prozessen, welche mit einer festen Frequenz f getaktet sind und bei jedem Takt ihren Zustand ändern können, wird $\Delta t = 1/f$ auf die Dauer eines Taktes gesetzt. Da **WILL_EXIST_NEXT** nur ein Makro für **WILL_EXIST** ist, gelten auch hier die schon gezeigten Bedingungen (1) – (3).

Definition: Prädikat **PREEXIST_PREVIOUS**

$$S^i \textbf{ PREEXIST\_PREVIOUS } ft \equiv S^i \textbf{ PREEXIST}[t_C-\Delta t, t_C]\, ft \equiv P_\pi(S^i, t_C-\Delta t, t_C, ft)$$

Erläuterung: Das Prädikat **PREEXIST_PREVIOUS** ist ein Makro für das Prädikat **PREEXIST** (siehe Abschnitt 4.5) und entstammt aus der direkten Übersetzung des Temporalen Operators Θx (war zum letzten Zeitpunkt beziehungsweise im letzten Zustand wahr, siehe Tabelle 1, Kapitel 4) und kann in der Temporalen Fuzzy-Logik mit kontinuierlicher und diskreter Zeit bei einer geeigneten Wahl von Δt verwendet werden. Bei der Auswertung des Prädikates interessiert nur ein kleiner Verlauf Δt in der Vergangenheit. Es ist zu beachten, dass Δt so gewählt werden muss, dass immer $\Delta t \leq t_C - t_A$ gilt. Bei Prozessen, welche mit einer festen Frequenz f getaktet sind und bei jedem Takt ihren Zustand ändern können, wird $\Delta t = 1/f$ auf die Dauer eines Taktes gesetzt. Da **PREEXIST_PREVIOUS** nur ein Makro für **PREEXIST** ist, gelten auch hier die schon gezeigten Bedingungen (1) – (3).

Definition: Prädikat **UNTIL**

$$S^i \textbf{ IS } ft_A \textbf{ UNTIL } [t_j, t_k, t_l]\; S^h \textbf{ IS } ft_B := P_{UNTIL}(S^i, S^h, t_j, t_k, t_l, ft_A, ft_B)$$

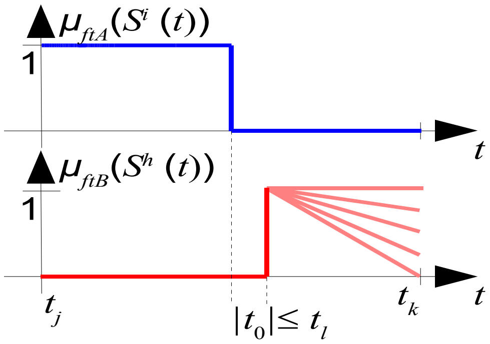
mit:

$$P_{reset}(t_{0,} S^i, t_j, t_k, ft_A) = \frac{1}{t_k - t_j}\left(\int_{t_j}^{t_0} P_\nu(i, t, ft_A)\,dt + \int_{t_0}^{t_k} 1 - P_\nu(i, t, ft_A)\,dt\right)$$

$$P_{edge}(t_{0,} S^h, t_j, t_k, ft_B) = \frac{1}{t_k - t_j}\left(\int_{t_j}^{t_0} 1 - P_\nu(h, t, ft_B)\,dt + (t_k - t_0)\cdot P_\nu(h, t_{0,} ft_B)\right)$$

$$t_0 \in \{t_0 | P_{reset}(t_0) = \max_t P_{reset}(t), t_0 \in [t_i, t_k]\}$$

$$P_{UNTIL}(S^i, S^h, t_j, t_k, t_l, ft_A, ft_B) = \min\left(P_{reset}(t_0, \ldots), \max_{|t_0 - t_m| \leq t_l} P_{edge}(t_m, \ldots)\right)$$



Erläuterung: Das Prädikat **UNTIL** testet mit dem Prädikat P_{reset} im Zeitintervall t_j bis t_k, ob ein Signalverlauf S^i, welcher zu Anfangs gültig war (in einer Zugehörigkeitsfunktion μ_{ftA} lag) durch das gültig werden des Signalverlaufes S^h (getestet mit P_{edge}) ungültig wurde und ungültig bleibt. Dabei ist bei dem Signalverlauf von S^h nur wichtig, dass er ungefähr zur selben Zeit ($\pm t_l/2$) gültig wird, wie S^i ungültig wird. Der weitere Verlauf von S^h spielt für den Aktivierungsgrad des Prädikates keine Rolle. Um P_{UNTIL} aus P_{reset} und P_{edge} zu berechnen kann entweder das Minimum oder das Produkt der beiden Prädikate genutzt werden. Gehen wir davon aus, dass die Zugehörigkeitsfunktionen Dreiecksfunktionen mit den Fuzzy Termen $ft_A = ft_B = \{(0,0), (0.5,1), (1,0)\}$ sind und die Zugehörigkeitsfunktionen demnach wie folgt definiert sind:

$$\mu_{ftA}(x)=\mu_{ftB}(x)=\begin{cases} 2x, 0<x\leq 0.5 \\ 1-2x, 0.5<x<1 \\ 0, sonst \end{cases}$$

Das Integral über die zweistelligen Funktionen für das Produkt $\mu_{ftA}(x)\cdot\mu_{ftB}(y)$ beziehungsweise für das Minimum $min(\mu_{ftA}(x), \mu_{ftB}(y))$ der beiden Zugehörigkeitsfunktionen ist 1/4 beziehungsweise 1/3. Berechnung der Integrale:

$$P_{Produkt}=\int_0^1\int_0^1 \mu_{ftA}(x)\cdot\mu_{ftB}(y)\,dy\,dx=4\cdot\int_0^{1/2}\int_0^{1/2} 2x\cdot 2y\,dy\,dx$$
$$=4\cdot\int_0^{1/2} 2x\cdot 2y|_0^{1/2}\,dx=\int_0^{1/2} 2x\,dx=1/4$$

$$P_{Minimum}=\int_0^1\int_0^1 min(\mu_{ftA}(x), \mu_{ftB}(y))\,dy\,dx$$
$$=4\cdot\int_0^{1/2}\int_0^{1/2} min(2x,2y)\,dy\,dx=8\cdot\int_0^{1/2}\int_{0,x\geq y}^{1/2} y\,dy\,dx+8\cdot\int_{0,y\geq x}^{1/2}\int_0^{1/2} x\,dy\,dx$$
$$=16\cdot\int_0^{1/2}\int_{0,x\geq y}^{1/2} y\,dy\,dx=16\cdot\int_0^{1/2}\int_0^{x} y\,dy\,dx=8\cdot\int_0^{1/2} x^2\,dx=1/3$$

Für das Prädikat liefert somit das Integral über die Zugehörigkeitsfunktionen mit dem Produkt eine kleinere Aktivierung als das mit dem Minimum. Aus diesem Grund favorisieren wir die Berechnung über das Minimum, da so das Prädikat **UNTIL** eher auf Signalverläufe mit den oben beschriebenen Eigenschaften P_{reset} und P_{edge} reagiert.

Da P_v normiert ist, also immer $0\leq P_v\leq 1$ gilt, ist das erste Integral von P_{reset} maximal $t_0 - t_j$ und das zweite Integral maximal $t_k - t_0$ groß. Da die Summe der beiden Integrale durch $t_k - t_j$ geteilt wird, ist P_{reset} maximal 1 und somit auch normiert. Für P_{edge} gilt dies auch, denn das erste Integral addiert mit $t_k - t_0$ mal dem normierten Prädikat P_v geteilt durch $t_k - t_j$ ist maximal 1. Da nun P_{edge} und P_{reset} normiert sind, ist dies auch deren Minimum beziehungsweise Produkt (1). Die Komposition von stetigen Funktionen mit der Minimumfunktion liefert immer stetige Funktionen, und da P_{edge} und P_{reset} stetig ist, ist somit auch P_{UNTIL} stetig (2). Für Bedingung (3) soll $\neg P_{UNTIL}=1-P_{UNTIL}$ gelten.

Diese Bedingung ist erfüllt, wenn Für $x{:=}P_{reset}$ und $y{:=}P_{edge}$ git, dass ist. Der Beweis:

$$\neg min(x,y)=\neg\begin{cases}x, x\leq y\\ y, sonst\end{cases}=\begin{cases}\neg x, x\leq y\\ \neg y, sonst\end{cases}$$
$$=\begin{cases}1-x, x\leq y\\ 1-y, sonst\end{cases}=1-\begin{cases}x, x\leq y\\ y, sonst\end{cases}=1-min(x,y)$$

Bleibt noch zu zeigen, dass $\neg P_{reset}=1-P_{reset}$ und $\neg P_{edge}=1-P_{edge}$ gilt.

$$\begin{aligned}
&\neg P_{reset}(t_{0,}S^i,t_j,t_k,ft_A)\\
&=\neg\frac{1}{t_k-t_j}\left(\int_{t_j}^{t_0}P_\nu(i,t,ft_A)\,dt+\int_{t_0}^{t_k}1-P_\nu(i,t,ft_A)\,dt\right)\\
&=\frac{1}{t_k-t_j}\left(\int_{t_j}^{t_0}\neg P_\nu(i,t,ft_A)\,dt+\int_{t_0}^{t_k}1-\neg P_\nu(i,t,ft_A)\,dt\right)\\
&=\frac{1}{t_k-t_j}\left(\int_{t_j}^{t_0}1-P_\nu(i,t,ft_A)\,dt+\int_{t_0}^{t_k}1-(1-P_\nu(i,t,ft_A))\,dt\right)\\
&=\frac{1}{t_k-t_j}\left(\int_{t_j}^{t_0}1\,dt-\int_{t_j}^{t_0}P_\nu(i,t,ft_A)\,dt+\int_{t_0}^{t_k}1\,dt-\int_{t_0}^{t_k}1-P_\nu(i,t,ft_A)\,dt\right)\\
&=\frac{t_k-t_j}{t_k-t_j}+\frac{1}{t_k-t_j}\left(-\int_{t_j}^{t_0}P_\nu(i,t,ft_A)\,dt-\int_{t_0}^{t_k}1-P_\nu(i,t,ft_A)\,dt\right)\\
&=1-\frac{1}{t_k-t_j}\left(\int_{t_j}^{t_0}P_\nu(i,t,ft_A)\,dt+\int_{t_0}^{t_k}1-P_\nu(i,t,ft_A)\,dt\right)\\
&=1-P_{reset}(t_{0,}S^i,t_j,t_k,ft_A)
\end{aligned}$$

$$\begin{aligned}
&\neg P_{edge}(t_{0,}S^h,t_j,t_k,ft_B)\\
&=\frac{1}{t_k-t_j}\left(\int_{t_j}^{t_0}1-\neg P_\nu(h,t,ft_B)\,dt+(t_k-t_0)\cdot\neg P_\nu(h,t_{0,}ft_B)\right)\\
&=\frac{1}{t_k-t_j}\left(\int_{t_j}^{t_0}1-(1-P_\nu(h,t,ft_B))\,dt+(t_k-t_0)\cdot(1-P_\nu(h,t_{0,}ft_B))\right)\\
&=\frac{1}{t_k-t_j}\left((t_0-t_j)+(t_k-t_0)+\int_{t_j}^{t_0}-1+P_\nu(h,t,ft_B)\,dt+(t_k-t_0)\cdot(-P_\nu(h,t_{0,}ft_B))\right)\\
&=1-\frac{1}{t_k-t_j}\left(\int_{t_j}^{t_0}1-P_\nu(h,t,ft_B)\,dt+(t_k-t_0)\cdot P_\nu(h,t_{0,}ft_B)\right)\\
&=1-P_{edge}(t_{0,}S^h,t_j,t_k,ft_B)
\end{aligned}$$

$\neg P_{edge} = 1 - P_{edge}$ gilt jedes beliebige t_0, insbesondere auch für $\max_{|t_0 - t_m| \le t_l} P_{edge}(t_m, \ldots)$. Somit ist Bedingung (3) bewiesen.

<u>Definition: Prädikat **SINCE**</u>

$$S^i \textbf{ IS } ft_A \textbf{ SINCE } [t_j, t_k, t_l]\ S^h \textbf{ IS } ft_B := P_{SINCE}(S^i, S^h, t_j, t_k, t_l, ft_A, ft_B)$$

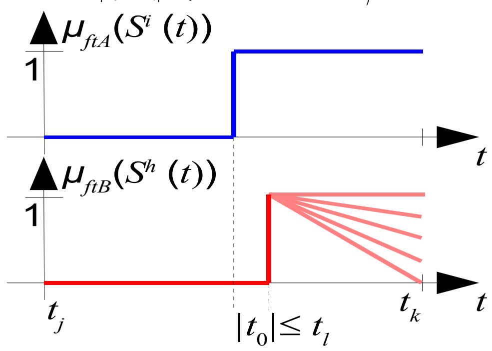
mit:

$$\neg P_{reset}(t_{0,} S^i, t_j, t_k, ft_A) = \frac{1}{t_k - t_j}\left(\int_{t_j}^{t_0} 1 - P_\nu(i, t, ft_A)\,dt + \int_{t_0}^{t_k} P_\nu(i, t, ft_A)\,dt\right)$$

$$P_{edge}(t_{0,} S^h, t_j, t_k, ft_B) = \frac{1}{t_k - t_j}\left(\int_{t_j}^{t_0} 1 - P_\nu(h, t, ft_B)\,dt + (t_k - t_0)\cdot P_\nu(h, t_{0,} ft_B)\right)$$

$$t_0 \in \{t_0 | \neg P_{reset}(t_0) = \max_t \neg P_{reset}(t), t_0 \in [t_i, t_k]\}$$

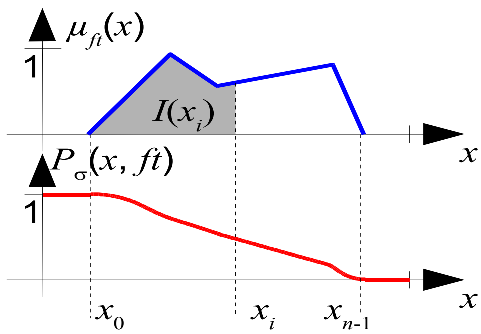
$$P_{SINCE}(S^i, S^h, t_j, t_k, t_l, ft_A, ft_B) = \min\left(\neg P_{reset}(t_0, \ldots), \max_{|t_0 - t_m| \le t_l} P_{edge}(t_m, \ldots)\right)$$



<u>Erläuterung:</u> Das Prädikat **SINCE** testet im Zeitintervall t_j bis t_k, ob ein Signalverlauf S^i, welcher zu Anfangs ungültig war (nicht in einer Zugehörigkeitsfunktion μ_{ft} lag) durch das gültig werden des Signalverlaufes S^h auch gültig wurde und gültig bleibt. Dabei ist bei dem Signalverlauf von S^h nur wichtig, dass er ungefähr zur selben Zeit $\pm t_l/2$ wie S^i gültig wird. Der weitere Verlauf von S^h spielt für den Aktivierungsgrad des Prädikates keine Rolle. Die Bedingungen (1) – (3) gelten für das Prädikat **SINCE** analog zu **UNTIL** mit den selben Begründungen.

<u>Definition: Prädikat **SMALLER**</u>

$$S^i \textbf{ SMALLER } ft := P_\sigma(x, ft) = 1 - \int_{-\infty}^{x} \mu_{ft}(y)\,dy / \int_{-\infty}^{\infty} \mu_{ft}(y)\,dy$$



<u>Erläuterung:</u> Die Prädikatenfunktion P_σ steht für *smaller*. Das Prädikat soll entscheiden, ob ein Wert x kleiner ist als ein gegebener Fuzzy-Term ft. Der Fuzzy-Term beginnt bei $(x_0, y_0) \in ft$ und endet bei $(x_{n-1}, y_{n-1}) \in ft$. Ist x echt kleiner als ft, also gilt $x<x_0$, so ist das Prädikat 1. Ist x jedoch echt größer als der ft, also gilt $x>x_{n-1}$, so ist das Prädikat 0. Liegt x jedoch innerhalb von ft, so wird das Prädikat umso kleiner, je näher man sich dem rechten Rand des Fuzzy-Terms nähert.

Zur genaueren Erläuterung definieren wir uns die Funktion $I(x) = \int_{-\infty}^{x} \mu_{ft}(y)\,dy$, welche eine Teilfunktion des Prädikates **SMALLER** ist. Sie liefert eine monoton steigende Funktion, denn die Zugehörigkeitsfunktion μ_{ft} ist nie negativ. $I(x)$ wird nun normiert und negiert, indem sie durch $I(\infty)$ geteilt und von 1 abgezogen wird. So entspricht die neu entstehende Funktion dem Prädikat **SMALLER**.

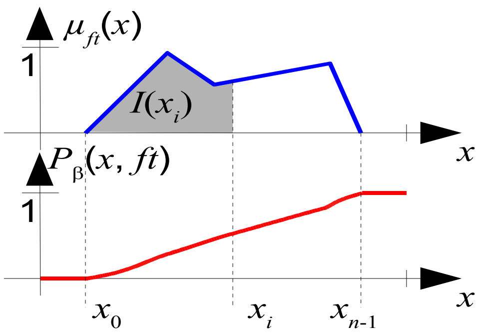
Es ist zu beachten, dass der Vergleich mit nur einem Fuzzy-Term gemacht wird. Es wird also keine Aussage über andere Fuzzy-Terme getroffen. Demnach müssen die Fuzzy-Terme nicht in geordneter Form gegeben sein. Sollten die Fuzzy-Terme jedoch geordnet sein, so lässt sich, wenn S_i **SMALLER** *ft* wahr ist, genau sagen, für welche Fuzzy-Terme S_i **SMALLER** *ft* auch noch wahr ist (und zwar für alle Fuzzy-Terme, welche größer als *ft* sind). Dies kann dazu genutzt werden um Bedingungen zu vereinfachen. Existieren zum Beispiel die Fuzzy-Terme *small*, *medium* und *big* für eine Variable *x*, so kann die Bedingung „(*x* **IS** *small*) **AND** (*x* **IS** *medium*) **AND** (*x* **IS NOT** *big*)” zu „(*x* **SMALLER** *big*)” abgekürzt werden. In der Regel besitzt eine Variable mehr als drei Fuzzy-Terme, wodurch die Ersparnis umso größer wird.

Für jedes beliebige x gilt immer, dass $0 \le \int_{-\infty}^{x} \mu_{ft}(y)\,dy \le \int_{-\infty}^{\infty} \mu_{ft}(y)\,dy$ ist, also ist der Bruch $\int_{-\infty}^{x} \mu_{ft}(y)\,dy / \int_{-\infty}^{\infty} \mu_{ft}(y)\,dy$ immer kleiner gleich 1 und größer gleich 0. Dies gilt auch für P_σ welches somit normiert ist (1). Auch gilt wiederum, dass die Komposition von stetigen Funktionen wieder stetige Funktionen liefert (2). Dass Bedingung (3) gilt wird bei der Definition des Prädikates **BIGGER** gezeigt.

<u>Definition: Prädikat **BIGGER**</u>

$$S^i\ \mathbf{BIGGER}\ ft := P_\beta(x, ft) = \int_{-\infty}^{x} \mu_{ft}(y)\,dy / \int_{-\infty}^{\infty} \mu_{ft}(y)\,dy$$

<u>Erläuterung:</u> Die Prädikatenfunktion P_β steht für *bigger*. Das Prädikat soll entscheiden, ob ein Wert x größer ist als ein gegebener Fuzzy-Term *ft*. Der Fuzzy-Term beginnt bei $(x_0, y_0) \in ft$ und endet bei $(x_{n-1}, y_{n-1}) \in ft$. Ist x echt kleiner als *ft*, also gilt $x<x_0$, so ist das Prädikat 0. Ist x jedoch echt größer als der *ft*, also gilt $x>x_{n-1}$, so ist das Prädikat 1. Liegt x jedoch innerhalb von *ft*, so wird das Prädikat, genau wie $I(x)$, umso größer, je näher man sich dem rechten Rand des Fuzzy-Terms nähert.



Eine genauere Erläuterung des Prädikates **BIGGER** kann analog beim Prädikate **SMALLER** nachgelesen werden, da sich die beiden Prädikate in ihrer Definition kaum unterscheiden. Die Bedingungen (1) und (2) gelten mit den selben Begründungen für das Prädikat **BIGGER** analog zum Prädikat **SMALLER**. Wie an den Graphen zu sehen ist, sind die Prädikate **BIGGER** und **SMALLER** bei einem gegebenem *ft* symmetrisch zur Achse $y = ½$. Die Bedingung (3) besagt, dass $\neg P_\beta = 1 - P_\beta = P_\sigma$ ist beziehungsweise dann auch $\neg P_\sigma = 1 - P_\sigma = P_\beta$ ist, also die Negation von **BIGGER** gleich **SMALLER** beziehungsweise die Negation von **SMALLER** gleich **BIGGER** ist. Beweise:

$$\neg P_{\beta}(x, ft) = 1 - P_{\beta}(x, ft) = 1 - \int_{-\infty}^{x} \mu_{ft}(y)\,dy / \int_{-\infty}^{\infty} \mu_{ft}(y)\,dy = P_{\sigma}(x, ft)$$

und

$$\neg P_{\sigma}(x, ft) = 1 - P_{\sigma}(x, ft) = 1 - \left(1 - \int_{-\infty}^{x} \mu_{ft}(y)\,dy / \int_{-\infty}^{\infty} \mu_{ft}(y)\,dy\right) = \int_{-\infty}^{x} \mu_{ft}(y)\,dy / \int_{-\infty}^{\infty} \mu_{ft}(y)\,dy = P_{\beta}(x, ft)$$

6. Anwendung

6.1. Wartungsbeispiel

Dieses Kapitel beschreibt ein komplettes Wartungsbeispiel geschrieben in der Sprache Temporal Fuzzy Control Language (TFCL), welche auf der Sprache Fuzzy Control Language (FCL, von [IEC97]) basiert. In einem Büroraum soll immer eine minimale Helligkeit vorherrschen, jedoch soll ein Maximalwert auch nicht überschritten werden. Eine Beschränkung ist, dass die Lampen etwa 15 Minuten benötigen, bis sie ihre maximale Helligkeit erreicht haben. Ist es im Büroraum zu dunkel, so müssen rechtzeitig zusätzliche Lampen eingeschaltet werden, denn wenn die minimale Helligkeit erreicht ist, dann ist es zu spät um weitere Lampen einzuschalten. Ist es jedoch zu Hell, weil die Sonneneinstrahlung zu stark ist, dann werden genügend Jalousien geschlossen um den Raum abzudunkeln. Aus diesen Gründen muss der Helligkeitsverlauf vorhergesagt werden. Die Aussenhelligkeit ändert sich im Laufe des simulierten Tages durch Sonnenauf- beziehungsweise Sonnenuntergang oder durch Wolken, welche die Sonnen verdunkeln. Wenn alle acht Lampen angeschaltet werden, wird mehr als genügend Licht produziert, auch wenn es Außen komplett dunkel ist. Auch genügen die drei Jalousien um die maximal mögliche Sonneneinstrahlung genügend zu verringern. Das TFCL Beispiel beinhaltet neben dem Wartungsteil auch noch einen Regelteil, um die Helligkeit im Büroraum auf einem konstanten Level zu halten.

Tabelle 2 beinhaltet die Beschreibung des Wartungsbeispieles geschrieben in TFCL. Die Systemvariablen *`numberOfLamps`* und *`shutterClosed`* repräsentieren die Anzahl der Lampen welche an zuschalten und Anzahl der Jalousien welche zu schließen sind und sind somit spezielle Ausgabevariablen. In der Eingabevariable *`brightness`* steht die gemessene Helligkeit im Büroraum. Die Ereignisvariablen *`maintenance`* und *`pWarning`* beinhalten Ereignisse, welche eintreten können. Die gemessene Helligkeit im Büroraum wird fuzzyfiziert. Es gibt hierzu fünf Fuzzy-Terme: *veryLow*, *low*, *med*, *high* und *veryHigh*. Dabei soll die Helligkeit größer als *veryLow* und *low* und kleiner als *high* und *veryHigh* sein, also ist ein Helligkeitswert von *med* erwünscht. Dies alles ist im ersten Abschnitt von Tabelle 2 beschrieben (alles oberhalb des Regelblockes **RULEBLOCK**).

Nach der Deklaration der Variablen und Fuzzy-Terme, werden die Regeln für den Fuzzy-Regler angegeben. Die *monitoring* und *prediction* Regel (Nummer 0) überprüft, ob die Helligkeit in der Zukunft einmal *veryLow* oder *low* wird. Wenn ja, wird das vorhergesagte Warnungsereignis *`brightness`* für die Helligkeit generiert. Die *maintenance* Regel (Nummer 1) überprüft, ob die Helligkeit schon eine viertel Stunde *low* ist und mehr als sieben Lampen eingeschaltet sind. Wenn ja, dann kann man daraus schließen, dass die Lampen nicht mehr genügend Helligkeit liefern, oder dass es defekte Lampen gibt. Es wird dann ein Wartungsereignis *replaceLamps* generiert, um den Benutzer anzuhalten, defekte Lampen auszutauschen. Die *control* Regeln (Nummer 2-6) werden genutzt um Lampen an oder auszuschalten oder um Jalousien zu öffnen oder zu schließen. Im Detail, schaltet Regel 2 Lampen an, wenn es in einer viertel Stunde zu dunkel sein würde und wenn alle Jalousien offen sind. Sollten Jalousien geschlossen sein, so möchten wir diese zuerst öffnen, bevor wir anfangen Lampen an zuschalten. Regel 3 öffnet eine Jalousie, wenn die Helligkeit in einer viertel Stunde zu dunkel ist und wenn mit der Bedingung „*shutterClosed* **PREEXIST** last quarter_hour *shutterClosed*" überprüft wurde, ob die Anzahl der geschlossenen Jalousien in der letzten viertel Stunde konstant war. Es wird also keine Jalousie geöffnet, wenn schon vor einer viertel Stunde eine geöffnet oder geschlossen wurde. Dadurch wird die Schaltfrequenz der Jalousien herunter gesetzt und so vermieden, dass ständig eine Jalousie geöffnet oder geschlossen wird. Wenn die Helligkeit zu hoch ist, dann werden mit Regel 4 und 5 Lampen ausgeschaltet, oder wenn alle Lampen ausgeschaltet sind, Jalousien geöffnet.

Wartungsbeispiel in TFCL

```
VAR_SYSTEM
  numberOfLamps actuator: Range: 0..8 REAL;
  shutterClosed actuator: Range: 0..2 REAL;
END_VAR
VAR_INPUT
  brightness: REAL;
END_VAR
VAR_EVENT
  maintenance;
  pWarning;
END_VAR_EVENT
FUZZIFY brightness
  TERM veryLow := (0, 1)(43, 1)(112, 0);
  TERM low := (43, 0)(112, 1)(128, 0);
  TERM med := (112, 0)(128, 1)(170, 0);
  TERM high := (128, 0)(170, 1)(213, 0);
  TERM veryHigh := (170, 0)(213, 1)(255,1);
  RANGE := (0 .. 255);
END_FUZZIFY
EVENT maintenance
  TASK replaceLamps;
END_EVENT
EVENT pWarning
  EVENT brightnessWarning;
END_EVENT
RULEBLOCK
  AND:MIN;
  OR:MAX;
  ACCU:MAX;
  ACT:MIN;
  PREDICTION:LINEARITY;
  RULE 0: IF (brightness WILL_EXIST veryLow) OR (brightness WILL_EXIST very-
  High) THEN (pWarning (brightnessWarning));
  RULE 1: IF (brightness WAS last quarter_hour veryLow) AND (numberOfLamps >
  7) THEN (maintenance (replaceLamps));
  RULE 2: IF ((brightness WILL_BE next quarter_hour veryLow) OR (brightness
  WILL_BE next quarter_hour low)) AND (shutterClosed < 0.5) THEN (numberO-
  fLamps ++);
  RULE 3: IF ((brightness WILL_BE next quarter_hour veryLow) OR (brightness
  WILL_BE next quarter_hour low)) AND (shutterClosed PREEXIST last
  quarter_hour shutterClosed) THEN (shutterClosed --);
  RULE 4: IF (brightness WILL_BE next quarter_hour high) OR (brightness
  WILL_BE next quarter_hour veryHigh) THEN (numberOfLamps --);
  RULE 5: IF (brightness WILL_BE next quarter_hour high) OR (brightness
  WILL_BE next quarter_hour veryHigh) AND (numberOfLamps < 0.5) THEN (shut-
  terClosed ++);
  END_RULEBLOCK
```

Tabelle 2: Komplettes Wartungsbeispiel geschrieben in der Sprache Temporal Fuzzy Control Language (TFCL) um die Helligkeit in einem Büroraum zu regeln und Wartungsaufträge zu generieren.

6.2. Experiment

Kapitel 6.1 führt ein Wartungsbeispiel ein, welches nun in einem simulierten Experiment eingesetzt wird. Die Randbedingungen sind, dass die Helligkeit im Büroraum nicht unter *low* oder *veryLow* und nicht über *high* oder *veryHigh* liegen. Die Helligkeit ist aus einem Intervall von 0 (Minimum) bis 255 (Maximum). Der Schwellwert, ab welcher Regelaktivierung die Regeln für die Systemvariablen *numberOfLamps* und *shutterClosed* feuern, ist auf 50% gesetzt. Das heißt, die Regeln 2-5 aus Tabelle 2 feuern nur bei einer Aktivierung der Bedingungen von mehr als 50%. Betrachtet man nun die Randbedingung an die Helligkeit, die Fuzzyfizierung der Helligkeit und den Schwellwert der Regelaktivierung, so bedeutet dies, dass die Helligkeit zwischen 120 und 149 liegen sollte und nicht die Helligkeit 170 überschreiten beziehungsweise 112 unterschreiten darf.

Die Helligkeit *o* von Außen, welche durch das Sonnenlicht verursacht ist und in den Büroraum einstrahlt hängt von der Tageszeit *t* ab und wird durch folgende Gleichung angenähert: $o(t) = 192 \cdot \sin\left(\frac{t-5h}{14h} \cdot \pi\right) + \text{noise}$

Wird eine Lampen eingeschaltet, so steigt deren abgestrahlte Helligkeit innerhalb von 15 Minuten linear von 0 auf 20 an. Wird die Lampe ausgeschaltet, so verringert sich die Helligkeit sofort auf 0.

Experimente mit der beschriebenen Simulation ergeben, dass die Helligkeit während des gesamten Tages innerhalb der geforderten Randbedingungen liegen. Bei der Helligkeit ist der Durchschnitt 135,4, das Maximum 163,4, das Minimum 113,8 und die Standardabweichung 6,9. Der Verlauf der Helligkeit ist in Abbildung 1 durch das Symbol X dargestellt.

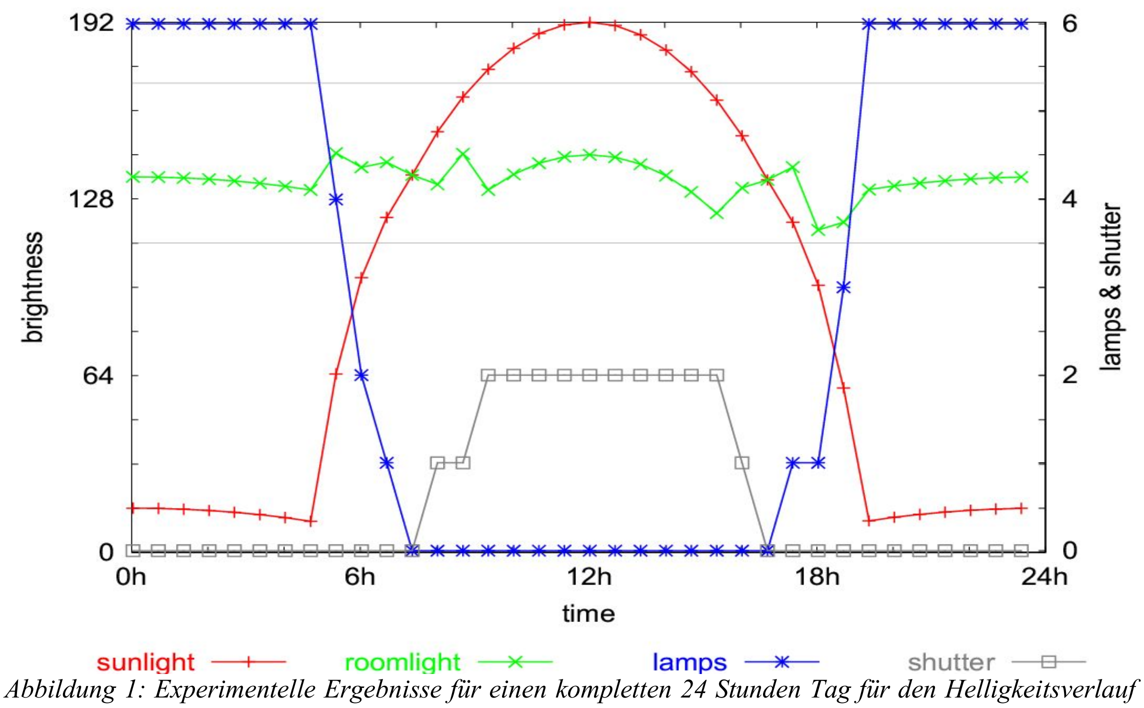


Abbildung 1: Experimentelle Ergebnisse für einen kompletten 24 Stunden Tag für den Helligkeitsverlauf im Büro (X), welcher durch Sonneneinstrahlung (+) gestört wird. Die Anzahl der angeschalteten Lampen (*) und geschlossenen Jalousien (□) regelt die Helligkeit zwischen dem maximal und minimal gewünschten Wert. Die maximal und minimal gewünschten Werte sind durch horizontale Linien bei Helligkeit gleich 112 und 149 dargestellt.

7. Schlussfolgerung

Wir haben gezeigt, dass es möglich ist, die Fuzzy-Logik mit temporalen Prädikaten zu erweitern, so dass wir die so genannte Temporale Fuzzy-Logik erhalten. Diese ermöglicht die Modellierung von zeitlichen Abhängigkeiten von Ereignissen und kann in einem Fuzzy-Regler zur Überwachung, Regelung und Wartung eingesetzt werden. Dieser Einsatz wird in einem Wartungsbeispiel gezeigt, in welchem ein Benutzer über defekte Lampen informiert wird.

Außerdem wurde im Hauptteil die mathematische Basis zu den temporalen Fuzzy-Prädikaten geschaffen. Die temporalen Fuzzy-Prädikate sind eine eins zu eins Abbildung der temporalen Prädikate, welche nach [Karjoth87] vollständig sind, das heißt mit ihnen können Bedingungen über den gesamten Zeitbereich erstellt werden. Somit sind die temporalen Fuzzy-Prädikate ebenfalls vollständig. Des weiteren kann deren Auswertung effizient ausgeführt werden.

Es ist möglich im Bedingungsteil einer Regel **AND** beziehungsweise **OR** verknüpfte Prädikate zu verwenden. Wie die Auswirkung von den angegebenen Zeiten auf die Schlussfolgerungen sind, ist zu diesem Zeitpunkt noch nicht bekannt, denn der Einfluss der Zeitangaben der Regelbedingung auf die Regelfolgerung ist noch nicht untersucht, aber er ist dennoch zu beachten. Im folgenden Beispiel werden die Daten vom Sensor S^i in der Vergangenheit aufzeichnet und damit ein Aktuator A^i in der Gegenwart gesteuert:

$$\textbf{IF } S^i \textbf{ WAS}[t_j,t_k]\, ft \textbf{ THEN } A^i \textbf{ IS } ft$$

Eine mögliche Interpretation ist, dass das Prädikat **IS** in einer Folgerung am wenigsten restriktiv ist, das heißt es ignoriert das Zeitintervall welches in der Bedingung beim Prädikat **WAS** gegeben ist und aktiviert die Regel, wenn die Bedingung erfüllt ist. Ersetzt man in der Folgerung **IS** durch **WAS**, so lässt man nur noch Zeitintervalle in der Vergangenheit zu und filtert Zeitintervalle in der Zukunft heraus, indem die entsprechenden Regeln nicht mehr aktiviert werden. **EXISTED** beziehungsweise **WILL_EXIST** lassen nur einen fixen Zeitpunkt in der Vergangenheit beziehungsweise Zukunft zu. Somit kann durch die Angabe eines temporalen Prädikates in der Folgerung einer Regel eine Beschränkung der Zeitintervalle der Bedingung bewirkt werden. Eine andere Möglichkeit der Interpretation ist, dass ein temporales Prädikat bei der Folgerung den Zeitpunkt der Aktivierung angibt. So dass ein Aktuator erst zu einem späteren Zeitpunkt gesetzt wird. Dies sind nur angedachte Ideen, welche vor allem bei der Verwendung mehrerer Prädikate pro Regelbedingung noch weiter untersucht werden müssen.

8. Literatur

[Althoff92] K.-D. Althoff, *Eine fallbasierte Lernkomponente als integrierter Bestandteil der MOLTKE-Werkbank zur Diagnose technischer Systeme*, Dissertation, Kaiserslautern, September 1992

[Bothe95] H.-H. Bothe, *Fuzzy-Logik – Einführung in Theorie und Anwendungen*, Springer-Lehrbuch, 2. Auflage, Berlin Heidelberg, 1995

[Bovenkamp98] E. G. P. Bovenkamp, J. C. A. Lubbe, *Temporal Reasoning with Fuzzy-Time-Objects*, 4th International Workshop on Temporal Representation and Reasoning, Daytona Beack, Florida, May 10-11, 1997

[Cárdenas02] M. A. Cárdenas Viedma, R. Martín Morales, *Syntax and Semantics for a Fuzzy Temporal Constraint Logic*, Annals of Mathematics and Artificial Intelligence, Volume 36, 2002

[Castillo02] O. Castillo, P. Melin, *A New Approach For Plant Monitoring Using Type-2 Fuzzy Logic and Fractal Theory*, Proceedings of the 5th International FLINS Conference, Belgium, Belgium, September 2002

[Fantoni00] P.F. Fantoni, M. Hoffmann, B. H. Nystad, *Integration of sensor validation in modern control room alarm systems*, Proceedings of the 4th International FLINS Conference, Belgium, August 2000

[Fick00] A. Fick, H. B. Keller, *Modellierung des Verhaltens Dynamischer Systeme mit erweiterten Fuzzyregeln*, Proceedings 10. Workshop Fuzzy Control des GMA-FA 5.22, Dortmund, Germany, 18. - 20. Oktober 2000

[Flender02] Flender Service GmbH, *Condition Monitoring for the highest Availability of Power Technology*, http://www.flender-cm.de/images/pdffiles/Leistungskurzschreibung_GB.pdf , Juni 2002

[Giron02] J. M. Giron-Sierra, G. Ortega, *A Survey of Stability of Fuzzy Logic Control with Aerospace Applications*, IFAC Proceedings of the 15th Triennial World Congress, Bacelona, Spain, 2002

[Haslum01] P. Haslum, *Models for Prediction*, IJCAI 2001 workshop on Planning under Uncertainty

[IEC97] IEC TC65/WG 7/TF8, *Fuzzy Control Programming*, International Technical Electronical Commission (IEC), 1997

[Karjoth87] G. Karjoth, *Prozeßalgebra und temporale Logik – angewandt zur Spezifikation und Analyse von komlexen Protokollen*, Diss. Mathematik/Informatik, Universität Stuttgart, 1987

[Lamine01] K. B. Lamine, F. Kabanza, *Reasoning About Robot Actions: A Model Checking Approach*, Revised Papers from the International Seminar on Advances in Plan-Based Control of Robotic Agents, Springer Verlag, 2001

[Palit00] A. K. Palit, *Artificial Intelligent Approaches to Times Series Forecasting*, Dissertation, Bremen, Januar 2000

[Škrjanc02] I. Škrjanc, D. Matko, *Fuzzy Predictive Functional Control in the State Space Domain, Journal of Intelligent and Robotic Systems*, Kluwer Academic Publishers, 2002

[Takagi85] T. Takagi, M. Suego, *Fuzzy identification of systems and its application to modeling and control*, IEEE Transactions on Systems, Man and Cybernetics, Vol. 15, No. 1, pp. 116-132, 1985.

[Watanabe86] H. Watanabe, *Schwerpunktsmethode*