

SIMFUZZY - ein in eine Simulationsumgebung integriertes Fuzzy Regelungsmodul

Ronald Ruzicka
SIMUTECH
Hadikgasse 150
A-1140 Wien

Abstract: Der Beitrag stellt eine Methode der Integration von Fuzzy Reglern in Simulationsprogramme vor, wie sie bei Sprachen vom CSSL Typ verwendet werden kann. Insbesondere unterstützt dieser Ansatz den Simulationspraktiker, der die Fuzzy Technik als ein Hilfsmittel und nicht unbedingt als Hauptzweck benützen möchte. Durch die Einbettung in die Simulationsumgebung wird ein hohes Maß an Flexibilität erreicht. Der Fuzzy Modul SIMFUZZY implementiert diese Methode in der Simulationsprache SIMUL_R.

1. Warum schon wieder ein neues Fuzzy System ?

Die Antwort ist einfach: weil die Simulationsanwender es wollten! So weit, so gut; aber was steckt dahinter?

Die Mehrheit aller derzeit am Markt befindlichen Fuzzy-Regelungsprogramme stellt dem Benutzer bequeme Möglichkeiten zur Definition seiner Fuzzy Regler zur Verfügung. Die Simulationsseite wird meist jedoch stark vernachlässigt. Die Kopplung der mit reinen Fuzzy Tools erzielten Ergebnisse mit bestehenden Simulationsprachen erfolgt i.a. nach einem der folgenden Verfahren:

- C-Code, der den Fuzzy Regler implementiert, wird erzeugt
- unter MS-Windows kann mittels DDE auf das Fuzzy System zugegriffen werden
- "zu Fuß" umsetzen der im Fuzzy System berechneten Resultate in die Zielsimulationssprache

C-Code wird von den meisten am Markt befindlichen Programmsystemen erzeugt [VALT94]. Es ist jedoch in den meisten Simulationssprachen relativ zeitaufwendig diesen Code zu inkludieren - insbesondere muß diese Integration nach Änderung des Fuzzy Reglers erneut durchgeführt werden.

Einige wenige Simulationssysteme erlauben die Kommunikation mittels DDE - ein zwar eleganter, aber sehr langsamer Mechanismus [REHF94].

Bei vielen Zielsimulationssprachen bleibt meist nur der dritte Weg, wie etwa in ACSL. Dieser ist meist mit einer völligen Neuimplementierung der Fuzzy Verfahren gleichzusetzen.

Der Simulationsspezialist geht jedoch von seinem Modell aus. Er möchte z.B. einen bisher verwendeten Regler durch einen Fuzzy Regler ersetzen. Er wünscht sich also ein Modul, das Fuzzy-Control einfach handhabbar in seine Simulationsmodelle inkludiert.

2. Ziele der Entwicklung eines integrierten Fuzzy Moduls

Die Wünsche des Simulationsspezialisten müssen die Ziele der Entwicklung einer integrierten Fuzzy Umgebung sein:

- Zugriff auf alle den Regler beschreibenden Größen von der gewohnten Simulationsoberfläche aus; d.h. die Reglerparameter stehen wie normale Modellvariable/konstanten zur Verfügung
- Optimierung der beschreibenden Größen - möglichst automatisch, mit vor-handenen Werkzeugen
- Möglichkeit der Anbindung von neuronalen Netzen - zur Reglerauslegung ("Neuro-Fuzzy")
- automatische Bedienbarkeit; z.B. durch eine Scriptsprache
- portierbare Formulierung der Regeln, Parameter und Resultate
- optional: Verwendbarkeit auf speziellen Zielprozessoren

Der Fuzzy Regler sollte im *Modell* wie ein normaler anderer Regler definiert und verwendet werden können.

Die Parameter und Regeln, die in der Reglerauslegungsphase einem interaktiven Prozeß unterworfen sind, sollten in der *Laufzeitumgebung* angebar sein.

3. SIMFUZZY - ein SIMUL_R Modul

In einem ersten Schritt wurden die oben genannten Bedingungen in dem Fuzzy Modul SIMFUZZY der Simulationssprache SIMUL_R zusammengefaßt [RUZI88].

Die textuelle Modellierungsseite von SIMUL_R entspricht dem CSSL Standard. Die Modelle werden übersetzt und zu einer Laufzeitumgebung ("Runtime-Interpreter") gebunden. Die Laufzeitumgebung ist mit sogenannten "Userdefined functions" erweiterbar. SIMFUZZY ist als solche implementiert.

3.1 Das Modell

Das Modell (Abbildung 1) enthält die Reglerdefinition mit Namen, Anzahl der Eingänge und Ausgänge, maximale Anzahl der linguistischen Variablen und der Stützpunkte der linguistischen Variablen in Form eines Makroaufrufes von *#fuzzy_define*.

Die linguistischen Variablen werden als Tabellenfunktionen beschrieben und, wie auch alle Reglerparameter, im Makro automatisch angelegt.

Der Regleraufruf erfolgt im Makro *#fuzzy_control* unter Angabe des Reglernamens und der Eingangs- und Ausgangsgrößen. Jeder Regler kann auch mehrmals aufgerufen werden.

```

#include 'fuzzy.def' " Makros definieren "

simfuzzy {
    ...
    #fuzzy_define(contr,3,2,5,10)
    " Regler definieren:  3 Eingänge,
                        2 Ausgänge "

    DYNAMIC {
        DERIVATIVE {
            #fuzzy_control(contr,t1,t2,x,y,z) "
Regleraufruf "
            x=...
            y=...
        }
    }
}

```

Abbildung 1: Fuzzy Reglerdefinition im Modell.

Der Fuzzy Regler kann ganz analog auch im graphischen Beschreibungs-werkzeug SIMDRAW angegeben werden.

Durch den *#fuzzy_define* Aufruf werden also alle benötigten Reglergrößen als Modellvariable angelegt. Sie können durch *contr...* angesprochen werden:

- Operationstypen
- Faktoren (z.B. Gamma)
- Zugehörigkeitsfunktionen als Tabellenfunktionen

3.2 Die Laufzeitumgebung

Im Runtime-Interpreter von SIMUL_R - und analog auch in einer entsprechenden Menü/Dialogoberfläche - werden sämtliche Reglerparameter gesetzt. Die Zugehörigkeitsfunktionen - also die Ordinaten der Tabellenfunktion - können graphisch editiert werden.

Bei textueller Eingabe im Interpreter wird - wie oben erwähnt - die userdefined function **SIMFUZZY** aufgerufen, deren Benutzung hierarchisch erfolgt: zuerst die Angabe des Reglernamens, dann der Name der Regel oder der linguistischen Variablen. Wird der Name weggelassen, so werden alle Regeln oder Variablen angesprochen.

SIMFUZZY bietet alle üblicherweise von einer Fuzzy-Umgebung gewünschten Eigenschaften, wie verschiedene Aggregat- (z.B. *fuzzyand*), Inferenz- und Akkumulationsstypen, vordefinierte Formen der Zugehörigkeitsfunktion (z.B. *triangle*) und Glättung/Kontrastierung (*dilate*).

Im Abbildung 2 sind einige Befehle beschrieben, die zum Regler in Abbildung 1 passen.

Die solchermaßen angegebenen Regeln und Parameter können auch auf einer Scriptdatei ("Commandfile") mittels *SIMFUZZY contr list* gesichert werden.

```

SIMFUZZY contr in i1,i2,i3;           Benennung der inputs und
SIMFUZZY contr out o1, o2;           outputs

SIMFUZZY contr variable l1,l2,l3,l4;  Benennung der linguist.
                                       Variablen

SIMFUZZY contr rule reg1 (1):(o1==l1)=(i1==l4)&&(i2==l2);
                                       Regeldefinition
SIMFUZZY contr rule reg1 aggregate fuzzyand(0.5), inference cut;
                                       Regel reg1: Aggregat- und
                                       Inferenztyp

SIMFUZZY contr rule accumulate maximum,
                                       Accumulation und Aus-
                                       gangsgrößenbest.
                                       output maximum;

SIMFUZZY contr variable l1 = triangle (0,1,3,4),
                                       setzt l1 als Dreieck,
                                       l3 = dilate, l4 = contrast;      glättet l3,
                                       kontrastiert l4

SIMFUZZY contr list ('zustand.txt');
                                       schreibt den contr Zustand als SIMFUZZY
                                       Befehle in zustand.txt

```

Abbildung 2: Laufzeitbefehle zur Reglerdefinition.

Dadurch, daß die Fuzzyparameter als Modellvariablen (z.B. der Aggregattyp, Γ -Wert) und die Zugehörigkeitsfunktionen als Tabellenfunktionen (Befehl LTABLE bei SIMUL_R) vorliegen, sind sie dem interaktiven Prozeß der Fuzzy-Regler-Auslegung (z.B. auch einer automatischen Optimierung mittels der SIMUL_R Funktionen OPTCONPAR oder GENOPT) zugänglich.

4. Zusammenfassung

Die hier vorgestellte Fuzzy Regler Methode und ihre Implementierung im Modul SIMFUZZY ist so konzipiert, daß sie in ähnlicher Form auch allen anderen (CSSL-) Simulationssprachen zur Verfügung gestellt werden kann. Sie gewährleistet dem Benutzer einer Simulationssprache eine optimale und flexible Integration in seine Simulationsumgebung.

5. Literatur

- [REHF94] U.Rehfueß: *Das Siemens Fuzzy-Entwicklungswerkzeug SieFuzzy*, Proc. 9. Symposium Simulationstechnik Stuttgart 1994, Vieweg, 1994
- [RUZI88] R.Ruzicka: *SIMUL_R - eine Simulationsprache mit speziellen Befehlen zur Modelldarstellung und -analyse*, Informatik Fachberichte 179, Proc. 5. Symposium Simulationstechnik Aachen, Springer, 1988
- [RUZI94] R.Ruzicka: *Optimierung technischer Systeme mittels Evolutionsstrategien - ein Standardverfahren in SIMUL_R*, Proc. 9. Symposium Simulationstechnik Stuttgart 1994, Vieweg, 1994
- [VALT94] v.Altrock, *Fuzzy Logic - Werkzeuge*, Oldenbourg, 1994