Regular Paper

重心法を用いた非ファジィ化計算の定義

On the Formalizations of Definition for Centroid Method as Defuzzification

三石 貴志 [1,*]
Takashi Mitsuishi[1,*]

1 長野大学企業情報学部
1 Faculty of Business and Informatics, Nagano University
658-1 Shimonogo, Ueda, Nagano, Japan

* takashi-mitsuishi@nagano.ac.jp

Proof checked by Mizar Version: 8.1.11 and MML Version: 5.68.1412
Received: April 13, 2022. Accepted: May 26, 2022.

Abstract

In this study, the centroid method which is one of the fuzzy inference processes is formulated. The centroid method is the most popular defuzzied method. By this method, defuzzified crisp value is obtained from domain of membership function as weighted average. Since the integral is used in centroid method, the integrability and bounded properties of membership functions are also mentioned. In this paper, the properties of piecewise linear functions consisting of two straight lines are mainly described.

1 はじめに

ファジィ推論は入力された前件部変数に対して，ファジィ集合（メンバシップ関数）が出力される．あいまい性を有した出力値では実用において不適当であるため出力のメンバシップ関数に非ファジィ化を行い，そのメンバシップ関数を表すのにもっとも適当なクリスプ値に変換する．非ファジィ化の方法でもっともよく用いられるのが重心法で他にもCenter of Sums 法，高さ法，密度モーメント法 [1–3] など多様に提案されている．本アーティクルでは重心法による非ファジィ化値を定義として形式化した．

ファジィ推論（Mamdani method，product-sum-gravity method）に使用される演算は主に数値と関数の積や頭切りといわれる MIN 演算，関数と関数との和算や MAX 演算などであるが [4, 5]，これらは Mizar Mathematical Library にてファジィ理論とは別に

形式化されている．稀にŁukasiewicz の演算（限界積）が用いられる場合がある．この演算は [6] にて形式化されている．そこで本研究では，これらの演算にて計算されたメンバシップ関数より重心法による非ファジィ化値を得るための諸性質，例えば定数倍や MIN・MAX 演算後の積分可能性（非ファジィ化値の計算可能性）などについて言及している．さらに通常の合成関数と異なる 2 つのグラフが交点でつながっている関数に関するいくつかの定理を形式化した．その具体例として区分的線形関数の非ファジィ化値を求める計算の形式化を行った．

2　メンバシップ関数の非ファジィ化

非ファジィ化法は重心法がもっとも広く用いられている．閉区間 A 上のメンバシップ関数 $\mu(x)$ の非ファジィ化値 x^* は，以下のように関数の定義域 A の値を関数値 $\mu(x)$ によって加重平均して定義域上の値を 1 つ求める方法である．

$$x^* = \frac{\int_A x\mu(x)dx}{\int_A \mu(x)dx}$$

本アーティクルでは，重心法によって非ファジィ化値を計算するために，閉区間 A 上の関数 f の重心値 $x^* =$ centroid(f,A) を以下のように定義した．

Listing 1. FUZZY_6:def 1

```
definition
 let A be non empty closed_interval Subset of REAL;
 let f be Function of REAL,REAL;
 assume that
 f is_integrable_on A and
 f | A is bounded;
 func centroid (f,A) -> Real equals :: FUZZY_6:def 1
 integral((id REAL)(#)f,A)/integral(f,A);
end;
```

$f(x) = x$ なる恒等写像は，Mizar Mathematical Library では様々な関数の形式によって表記されているが，本アーティクルでは以下の関数を用いた [7]．

Listing 2. RELAT_1:def 10

```
definition
 let X be set ;
 func id X -> Relation means :: RELAT_1:def 10
 for x, y being object holds
 ( [x,y] in it iff ( x in X & x = y ) );
end;
```

Listing 1 では，重心を求める関数の値域を $[0, 1]$ に限定しておらずメンバシップ関数でないものにも適用できる．

$b \in [a, c]$ とする．閉区間 $[a, c]$ の部分集合 $[b, c]([a, b])$ において $f(x) = 0$ であるとき，この区間には重心値が存在しない．したがって以下の定理は，その区間を分断して非ファジィ化値を計算する区間を $[a, b]([b, c])$ とすることができることを表している．

Listing 3. FUZZY_6:8, FUZZY_6:9

```
theorem :: FUZZY_6:8
for a,b,c be Real, f be Function of REAL,REAL st
a < b & b <= c & f is_integrable_on ['a,c'] & f | ['a,c'] is bounded & for x be Real st x in ['b,c'] holds f.x = 0 holds
centroid(f,['a,c']) = centroid(f,['a,b']);

theorem :: FUZZY_6:9
for a,b,c be Real, f be Function of REAL,REAL st
a <= b & b < c & f is_integrable_on ['a,c'] & f | ['a,c'] is bounded & for x be Real st x in ['a,b'] holds f.x = 0 holds
centroid(f,['a,c']) = centroid(f,['b,c']);
```

上記定理の証明において，以下の定理 [8] が有用な道具となった.

Listing 4. INTEGRA6:17, INTEGRA6:18

```
theorem :: INTEGRA6:17
for a, b, c being Real for f being PartFunc of REAL,REAL st
a <= b & f is_integrable_on ['a,b'] & f | ['a,b'] is bounded & ['a,b'] c= dom f & c in ['a,b'] holds
( f is_integrable_on ['a,c'] & f is_integrable_on ['c,b'] & integral (f,a,b) = (integral (f,a,c)) + (integral (f,c,b)) );

theorem :: INTEGRA6:18
for a, b, c, d being Real for f being PartFunc of REAL,REAL st
 a <= c & c <= d & d <= b & f is_integrable_on ['a,b'] & f | ['a,b'] is bounded & ['a,b'] c= dom f holds
( f is_integrable_on ['c,d'] & f | ['c,d'] is bounded & ['c,d'] c= dom f );
```

以下の定理は非ファジィ化関数を定数倍しても非ファジィ化値は変化しないことを示している.

Listing 5. FUZZY_6:32

```
theorem :: FUZZY_6:32
for r be Real, f be Function of REAL,REAL st r <> 0 & f is_integrable_on A & f | A is bounded holds
centroid((r (#) f),A) = centroid(f,A);
```

3 重心法のためのメンバシップ関数の積分に関する諸性質

メンバシップ関数の重心法による非ファジィ化値の存在に関連する性質として，閉区間 A において関数 f の積分値が正の値であれば，その区間に $f(c) > 0$ となる $c \in A$ が存在する定理を証明した.

Listing 6. FUZZY_6:10

```
theorem :: FUZZY_6:10
for f being Function of REAL,REAL st f is_integrable_on A & f | A is bounded & integral(f,A) > 0 holds
ex c being Real st c in A & f.c > 0;
```

Mamdani 推論法ではメンバシップ関数同士の MAX 演算が用いられ，前件部変数をファジィ数とした推論法 [9] ではメンバシップ関数同士の MIN 演算が用いられる．下記の定理は，関数同士の MAX・MIN 演算を積，和および絶対値の演算として表した.

Listing 7. FUZZY_6:12-15

```
theorem :: FUZZY_6:12
for f,g be Function of REAL,REAL holds min(f,g) = (1/2) (#) ((f + g) − abs(f − g));

theorem :: FUZZY_6:13
for f,g be Function of REAL,REAL st f is_integrable_on A & f | A is bounded & g is_integrable_on A & g | A is bounded
holds min(f,g) is_integrable_on A & min(f,g) | A is bounded &
integral(min(f,g),A) = (1/2)∗( integral(f,A) + integral(g,A) − integral(abs(f−g),A) );

theorem :: FUZZY_6:14
for f,g be Function of REAL,REAL holds max(f,g) = (1/2) (#) (f + g + abs(f − g));

theorem :: FUZZY_6:15
for f,g be Function of REAL,REAL st f is_integrable_on A & f | A is bounded & g is_integrable_on A & g | A is bounded
holds max(f,g) is_integrable_on A & max(f,g) | A is bounded &
integral(max(f,g),A) = (1/2)∗( integral(f,A) + integral(g,A) + integral(abs(f−g),A) );
```

以下の定理はファジィ推論において，前件部適合度を後件部のメンバシップ関数に反映させた後のメンバシップ関数の積分可能性を表し，その関数の重心が得られることを意味する．この定理は今後Łukasiewicz の演算（限界積）を用いた場合にも適用可能である．

Listing 8. FUZZY_6:17

```
theorem :: FUZZY_6:17
for r1,r2 be Real, f,F be Function of REAL,REAL st
f is_integrable_on A & f | A is bounded & for x being Real holds F.x = min(r1, r2∗(f.x)) holds
F is_integrable_on A & F | A is bounded;
```

4 区分的線形関数の非ファジィ化値

以下の定理は Lipschitz 連続な関数 f および g がある点 c において等しい値をとるとき，それらが c において結合した関数は再び Lipschitz 連続になることを示している．また一次関数の Lipschitz 連続性を，自明ではあるが今後の関連項目の証明のために形式化した．

Listing 9. FUZZY_6:26, FUZZY_6:27

```
theorem :: FUZZY_6:26 :::Lipschitzian
for c being Real, f,g,F be Function of REAL,REAL st
f is Lipschitzian & g is Lipschitzian & f . c = g . c & F = (f | ].−infty,c.[) +∗ (g | [.c,+infty.[) holds F is Lipschitzian;

theorem :: FUZZY_6:27
for a,b being Real holds (AffineMap (a,b)) is Lipschitzian;
```

Lsting 10 は，以下に示す傾きの異なる 2 本の直線が交点 $\dfrac{q-b}{a-p} \in A$ で結合している区分線形関数

$$f(x) = \begin{cases} ax+b & \left(x \leq \dfrac{q-b}{a-p}\right) \\ px+q & \left(x \geq \dfrac{q-b}{a-p}\right) \end{cases}$$

の閉区間 A における重心

$$x^* = \frac{\int_A xf(x)dx}{\int_A f(x)dx} = \frac{\frac{a}{3}(t^3 - s^3) + \frac{b}{2}(t^2 - s^2) + \frac{p}{3}(u^3 - t^3) + \frac{q}{2}(s^2 - t^2)}{\frac{a}{2}(t^2 - s^2) + b(t - s) + \frac{p}{2}(u^2 - t^2) + q(u - t)}$$

を表している．ただし，$a \neq p$ であり，$s =$ lower_ bound A は A の下界，$t = \dfrac{q - b}{a - p}$，$u =$ upper_ bound A は A の上界である．

Listing 10. FUZZY_6:48

```
theorem :: FUZZY_6:48
for a,b,p,q,c,d,e being Real, f be Function of REAL,REAL st
a <> p & f | A = AffineMap (a,b) | [.lower_bound A,(q-b)/(a-p).] +* AffineMap (p,q) | [.(q-b)/(a-p),upper_bound A.]
& (q-b)/(a-p) in A holds
centroid(f,A) = ( 1/3*a*(((q-b)/(a-p))^3 - (lower_bound A)^3) + 1/2*b*(((q-b)/(a-p))^2 - (lower_bound A)^2)
  + 1/3*p*((upper_bound A)^3 - ((q-b)/(a-p))^3) + 1/2*q*((upper_bound A)^2 - ((q-b)/(a-p))^2) )
/ ( 1/2*a*(((q-b)/(a-p))^2 -(lower_bound A)^2) + b*((q-b)/(a-p) - lower_bound A)
  + 1/2*p*((upper_bound A)^2 -((q-b)/(a-p))^2) + q*(upper_bound A -(q-b)/(a-p)) );
```

5 まとめ

本研究では，ファジィ推論における非ファジィ化の一つ重心法の計算について形式化を行った．メンバシップ関数に対して適用される重心法ではあるが，形式化においてはその対象を値域が区間 $[0, 1]$ の関数に限定していない．ファジィ推論の出力であるメンバシップ関数は，複数のメンバシップ関数の MAX 演算や和算による結合である場合が多いので，重心法の対象となるべく，2 つの関数が結合した関数の諸性質に多くの紙面を割いた．それに伴って，重心計算の分子の非積分関数である恒等写像とメンバシップ関数の積の積分可能性なども形式化された．本研究で形式化した重心法は SIRMs 推論法 [10] の非ファジィ化にも適用が可能である．アーティクルの最後の定理として，2 本の直線が結合した区分的線形関数の非ファジィ化値を示した．本稿以降のアーティクルでは，非線形関数に重心法を適用する計算の形式化を予定している．将来は重心法のメンバシップ関数集合の上で定義された汎関数としての連続性に関する性質を形式化し，筆者らが証明したファジィ最適制御の存在性 [11] の検証を行う．

参考文献

[1] ROSS TJ. Fuzzy Logic With Engineering Application. John Wiley and Sons Ltd; 2010.

[2] Van Leekwijck W, Kerre EE. Defuzzification: criteria and classification. Fuzzy sets and systems. 1999;108(2):159–178.

[3] KATAFUCHI T, ASAI K, FUJITA H. Investigation of Deffuzification in Fuzzy Inference : Proposal of a New Defuzzification Method (in Japanese). Medical Imaging and Information Sciences. 2001;18(1):19–30.

[4] MAMDANI EH. Application of fuzzy algorithms for control of simple dynamic plant. Proc IEE. 1974;121(12):1585–1588.

[5] Mizumoto M. Improvement of fuzzy control (IV)-Case by product-sum-gravity method. In: Proc. 6th Fuzzy System Symposium, 1990; 1990. p. 9–13.

[6] Grabowski A. Basic Formal Properties of Triangular Norms and Conorms. Formalized Mathematics. 2017;25(**2**):93–100.

[7] Woronowicz E. Relations and Their Basic Properties. Formalized Mathematics. 1990;1(**1**):73–83. Available from: `http://fm.mizar.org/1990-1/pdf1-1/relat_1.pdf`.

[8] Endou N, Shidama Y, Yamazaki M. Integrability and the Integral of Partial Functions from $\mathbb{R}$ into $\mathbb{R}$. Formalized Mathematics. 2006;14(**4**):207–212.

[9] Mitsuishi T, Terashima T, Shimada N, Homma T, Shidama Y. SIRMs fuzzy approximate reasoning using L-R fuzzy number as premise valuable. In: 2013 8th International Conference on System of Systems Engineering; 2013. p. 25–27.

[10] Seki H, Ishii H, Mizumoto M. On the Generalization of Single Input Rule Modules Connected Type Fuzzy Reasoning Method. IEEE Transactions on Fuzzy Systems. 2008;16(5):1180–1187.

[11] Mitsuishi T, Shidama Y. Compactness of family of fuzzy sets in L^2 space with application to optimal control. IEICE transactions on fundamentals of electronics, communications and computer sciences. 2009;92(4):952–957.

Mizar article information

Works in Progress

FUZZY_6 Centroid Method for Defuzzification
by Takashi Mitsuishi

Summary: Fuzzy inference method outputs a fuzzy set (membership function) for a given premise variable. Membership functions are unsuitable for practical use and are converted to crisp values by defuzzification process. The centroid method is the most used method. Various other defuzzification methods such as the Center of Sums method, the height method and the density moment method are proposed [1–3]. In this article, we present formalization of defuzzified crisp value of membership function using centroid method. And related theorems are formalized.

The operations used in Mamdani method and product-sum-gravity method are simple and have already been formalized in Mizar Mathematical Library. Moreover Łukasiewicz operations are in the article [6]. On the other, the integrability and bounded theorems of the membership functions changed by these operations are described for

defuzzified value by centroid method. As a concrete example, we finally formalized the calculation of defuzzified value of the piecewise linear function.

Listing 11. FUZZY_6 - abstract

```
environ

 vocabularies NUMBERS, XBOOLE_0, SUBSET_1, XXREAL_1, CARD_1, RELAT_1, TARSKI,
      FUNCT_1, XXREAL_0, PARTFUN1, ARYTM_1, ARYTM_3, COMPLEX1, FUZZY_1,
      MEASURE5, FUZNUM_1, REAL_1, ORDINAL2, XXREAL_2, JGRAPH_2, FUNCT_4,
      FCONT_1,SQUARE_1, VALUED_1,FDIFF_1, INTEGRA1,INTEGRA5,SEQ_4, NAT_1,
      PREPOWER,NEWTON,RFUNCT_3, CARD_3, MEASURE7, FINSEQ_1, POLYEQ_3, FUZZY_6;

 notations TARSKI, XBOOLE_0, SUBSET_1, NEWTON, RELAT_1, ORDINAL1, NUMBERS,
      SQUARE_1, BINOP_1, NORMSP_1,METRIC_1, INT_1, XCMPLX_0,XXREAL_0,
      COMPLEX1,XXREAL_2,XREAL_0, REAL_1, FUNCT_1, PBOOLE, RELSET_1,
      MEASURE5, PARTFUN1, FUNCT_2,FUNCT_3,INTEGRA1,INTEGRA2,RFUNCT_3,
      FUNCOP_1, NAT_1, VALUED_0, VALUED_1, FUNCT_4,
      ARYTM_2, ARYTM_1, ARYTM_0, COMSEQ_2, FDIFF_1,
      MEMBERED, FINSEQ_1, FINSEQ_2, SEQ_2,SEQ_4, PREPOWER,
      FCONT_1, SIN_COS, PARTFUN2, RFUNCT_1, POLYEQ_3,
      RVSUM_1,RCOMP_1, TOPMETR, TAYLOR_1, PRE_TOPC, TOPS_2, POLYDIFF,
      INTEGRA5,COUSIN2, DOMAIN_1, TSEP_1, TMAP_1,
      FUZZY_1, BORSUK_1, FUZNORM1,FUZNUM_1;


 theorems
      XBOOLE_0,TARSKI, FUNCT_1, ABSVALUE, PARTFUN1, TAYLOR_1,
      COMPLEX1, XREAL_1, XXREAL_0, XXREAL_1, FUNCT_2, RELAT_1,
      XREAL_0, FCONT_1, XCMPLX_1, FUZZY_2, FUNCT_4, SUBSET_1,
      FDIFF_1, VALUED_1, INTEGRA1, INTEGRA5, PREPOWER, NEWTON,
      INTEGRA6, SEQ_4, RFUNCT_3, INTEGRA2, INTEGRA9, COMSEQ_2,COUSIN2,RFUNCT_1,
      FINSEQ_3, POLYEQ_3;

 constructors
      XBOOLE_0, ZFMISC_1, SUBSET_1, FUNCT_1, FUNCT_2, XCMPLX_0,
      XXREAL_0, XREAL_0, COMPLEX1, RFUNCT_1, SEQ_4, RELSET_1, FUZZY_1, FCONT_1,
      FUNCT_4,FUZNUM_1,NUMPOLY1,LFUZZY_1,RELAT_1,SIN_COS,XXREAL_3,NUMBERS,
      SIN_COS6,TOPMETR,SQUARE_1,TAYLOR_1,POWER, COMPTS_1, RCOMP_1,
      FINSEQ_1, FINSEQ_2, INT_1,ORDINAL1,
      FDIFF_1, NORMSP_1,METRIC_1,VALUED_1,PARTFUN1,MEASURE5,
      COMSEQ_2, PARTFUN2,POLYDIFF,PREPOWER,NEWTON,
      RFUNCT_3,FUZNORM1, BINOP_1, BINOP_2,INTEGRA1,INTEGRA2,INTEGRA3,INTEGRA5,
      COUSIN2,SEQ_1,SEQ_2,RVSUM_1, NAT_1,
      XXREAL_2, EXTREAL1, MESFUNC5,TOPS_2, CONNSP_1,
      TMAP_1, PCOMPS_1,REAL_1,ARYTM_1,ARYTM_0,PBOOLE,FUNCOP_1,FUNCT_3,
      POLYEQ_3;

 registrations RELSET_1, NUMBERS, XXREAL_0, MEMBERED, XBOOLE_0, VALUED_0,
      VALUED_1,COMSEQ_2,PDIFFEQ1,FDIFF_1,FUZZY_1,
      FUNCT_2,XREAL_0, ORDINAL1, FCONT_1, RELAT_1, TOPREALB, FUNCT_4, FUNCT_1,
      XCMPLX_0, NAT_1, RCOMP_1,FUZNUM_1,NUMPOLY1,LFUZZY_1,SIN_COS,XXREAL_3,
      SIN_COS6,TOPMETR,SQUARE_1,SIN_COS3, CARD_3, INT_1,SUBSET_1,SIN_COS9,
      NORMSP_1,PARTFUN1,INTEGRA1,MEASURE5,COMSEQ_1,
      FDIFF_2,POLYDIFF,PREPOWER,NEWTON,RFUNCT_3,FUZNORM1,INTEGRA2,RFUNCT_1,
      SEQ_2,SEQ_1,RVSUM_1,NAT_3, PRE_TOPC, METRIC_1, BORSUK_1, CONNSP_1,
      EXTREAL1, SUPINF_1, SUPINF_2;

 requirements NUMERALS, REAL, SUBSET, BOOLE, ARITHM ;

 definitions XBOOLE_0, TARSKI, XXREAL_2,FUZNUM_1,NUMPOLY1,FUZZY_1,RELAT_1,
      XXREAL_3,NUMBERS,SIN_COS6,TOPMETR,SQUARE_1, COMPTS_1, RCOMP_1,
      FCONT_1,INT_1,SUBSET_1,VALUED_1,XXREAL_0,SIN_COS,
      PARTFUN1,INTEGRA1,INTEGRA2,INTEGRA3,INTEGRA5,MEASURE5,FUNCT_2,RFUNCT_1,
      BINOP_1,FUZNORM1,COMSEQ_2,FINSEQ_1,NAT_1,TAYLOR_1;

 equalities FUZZY_1, RCOMP_1,FUZNUM_1,NUMPOLY1,RELAT_1,XXREAL_3,
      NUMBERS,SIN_COS6,SQUARE_1,FUNCT_4,RFUNCT_1;
 expansions TARSKI, FCONT_1,FUZNUM_1,NUMPOLY1,FUZZY_1,RELAT_1,XXREAL_3,
```

```
    NUMBERS,SQUARE_1,FUNCT_4,RFUNCT_1;

begin

reserve A for non empty closed_interval Subset of REAL;


definition
 let A be non empty closed_interval Subset of REAL;
 let f be Function of REAL,REAL;
 assume that
 f is_integrable_on A and
 f | A is bounded;
 func centroid (f,A) -> Real equals
:: FUZZY_6:def 1
 integral((id REAL)(#)f,A)/integral(f,A);
end;

theorem :: FUZZY_6:1
for a,b,c be Real st a < b & c > 0
holds centroid (AffineMap (0,c),['a,b']) = (a+b)/2;

theorem :: FUZZY_6:2
for a,b be Real holds
(id REAL) is_integrable_on ['a,b'] & (id REAL) | ['a,b'] is bounded;

theorem :: FUZZY_6:3
 (id REAL) is_integrable_on A & (id REAL) | A is bounded;

theorem :: FUZZY_6:4
for e being Real
for f being PartFunc of REAL,REAL st
A c= dom f & ( for x being Real st x in A holds f . x = e ) holds
( f is_integrable_on A & f | A is bounded &
integral (f,(lower_bound A),(upper_bound A))
        = e * ((upper_bound A) - (lower_bound A)) );

theorem :: FUZZY_6:5
for f be Function of REAL,REAL st
(for x be Real st x in A holds f.x = 0) holds
integral(f,A) = 0;

theorem :: FUZZY_6:6
for f be Function of REAL,REAL st
f is_integrable_on A & f | A is bounded
holds
(id REAL) (#) f is_integrable_on A &
((id REAL) (#) f) | A is bounded;

theorem :: FUZZY_6:7
for a,b,c be Real st a < b holds
['a,b'] c= [#]REAL & lower_bound ['a,b'] = a & upper_bound ['a,b'] = b;

theorem :: FUZZY_6:8
for a,b,c be Real, f be Function of REAL,REAL st
a < b & b <= c &
f is_integrable_on ['a,c'] & f | ['a,c'] is bounded &
for x be Real st x in ['b,c'] holds f.x = 0
holds
centroid(f,['a,c']) = centroid(f,['a,b']);

theorem :: FUZZY_6:9
for a,b,c be Real, f be Function of REAL,REAL st
a <= b & b < c & f is_integrable_on ['a,c'] & f | ['a,c'] is bounded &
for x be Real st x in ['a,b'] holds f.x = 0
holds
centroid(f,['a,c']) = centroid(f,['b,c']);
```

```
theorem :: FUZZY_6:10
for f being Function of REAL,REAL st
f is_integrable_on A & f | A is bounded & integral(f,A) > 0
holds
ex c being Real st c in A & f.c > 0;

theorem :: FUZZY_6:11
for r be Real, f be FuzzySet of REAL, F be Function of REAL,REAL st
r > 0 & f is_integrable_on A & f | A is bounded &
for x being Real holds F.x = min(r, f.x)
holds
integral(F,A) >= 0;

theorem :: FUZZY_6:12
for f,g be Function of REAL,REAL holds
min(f,g) = (1/2) (#) ((f + g) - abs(f - g));

theorem :: FUZZY_6:13
for f,g be Function of REAL,REAL st
f is_integrable_on A & f | A is bounded &
g is_integrable_on A & g | A is bounded
holds
min(f,g) is_integrable_on A & min(f,g) | A is bounded &
integral(min(f,g),A)
 = (1/2)*( integral(f,A) + integral(g,A) - integral(abs(f-g),A) );

theorem :: FUZZY_6:14
for f,g be Function of REAL,REAL holds
max(f,g) = (1/2) (#) (f + g + abs(f - g));

theorem :: FUZZY_6:15
for f,g be Function of REAL,REAL st
f is_integrable_on A & f | A is bounded &
g is_integrable_on A & g | A is bounded
holds
max(f,g) is_integrable_on A & max(f,g) | A is bounded &
integral(max(f,g),A)
= (1/2)*( integral(f,A) + integral(g,A) + integral(abs(f-g),A) );

theorem :: FUZZY_6:16
for r1,r2 be Real, f be Function of REAL,REAL st
f is_integrable_on A & f | A is bounded
holds
min(AffineMap(0,r1),(r2 (#) f)) is_integrable_on A &
min(AffineMap(0,r1),(r2 (#) f)) | A is bounded;

theorem :: FUZZY_6:17
for r1,r2 be Real, f,F be Function of REAL,REAL st
f is_integrable_on A & f | A is bounded &
for x being Real holds F.x = min(r1, r2*(f.x))
holds
F is_integrable_on A & F | A is bounded;

theorem :: FUZZY_6:18
for s be Real, f,g being Function of REAL,REAL holds
( f | ].-infty,s.[ ) +* ( g | [.s,+infty.[ )
is Function of REAL,REAL;

theorem :: FUZZY_6:19
for a,b,c being Real, f,g,F be Function of REAL,REAL st
a <= b & b <= c &
F = f | [.a,b.] +* g | [.b,c.]
holds
F is Function of ['a,c'],REAL;

theorem :: FUZZY_6:20
for a,b,c being Real, f,g,F be Function of REAL,REAL st
a <= b & b <= c & F = f | [.a,b.] +* g | [.b,c.]
```

holds
F = F | [. a,c .];

theorem *:: FUZZY_6:21*
for a,b,c **being** Real, f,g,h **be** Function **of** REAL,REAL **st**
a <= b **&** b <= c **&**
f | ['a,c'] **is** bounded **&** g | ['a,c'] **is** bounded **&**
h = (f| [.a,b.]) +* (g | [.b,c.]) **&** f.b = g.b
holds
h | ['a,c'] **is** bounded;

theorem *:: FUZZY_6:22*
for a,b,c **being** Real, f,g,h **be** Function **of** REAL,REAL **st**
a <= b **&** b <= c **&**
f | ['a,c'] **is** bounded **&** g | ['a,c'] **is** bounded **&**
h | [.a,c.] = (f| [.a,b.]) +* (g | [.b,c.]) **&** f.b = g.b
holds
h | ['a,c'] **is** bounded;

theorem *:: FUZZY_6:23*
for c **being** Real, f,g **be** Function **of** REAL,REAL **st**
f | A **is** bounded **&** g | A **is** bounded
holds
((f |].−infty,c.[) +* (g | [.c,+infty.[)) | A **is** bounded;

theorem *:: FUZZY_6:24*
for a,b,c **be** Real, f,g,h,F **be** Function **of** REAL,REAL **st**
a <= b **&** b <= c **&** f **is** continuous **&** g **is** continuous **&**
h | [.a,c.] = (f | [.a,b.]) +* (g | [.b,c.]) **&** f.b = g.b **&** F = h | [.a,c.]
holds
F **is** continuous;

theorem *:: FUZZY_6:25*
for A **being non** empty closed_interval Subset **of** REAL,
f **be** Function **of** REAL,REAL **st**
f **is** continuous **holds** f is_integrable_on A **&** f | A **is** bounded;

theorem *:: FUZZY_6:26*
for c **being** Real, f,g,F **be** Function **of** REAL,REAL **st**
f **is** Lipschitzian **&** g **is** Lipschitzian **&**
f . c = g . c **&** F = (f |].−infty,c.[) +* (g | [.c,+infty.[)
holds F **is** Lipschitzian;

theorem *:: FUZZY_6:27*
for a,b **being** Real **holds** (AffineMap (a,b)) **is** Lipschitzian;

theorem *:: FUZZY_6:28*
for a,b,p,q **being** Real, f **be** Function **of** REAL,REAL **st** a <> p **&**
f = (AffineMap (a,b) |].−infty,(q−b)/(a−p).[ )
+* (AffineMap (p,q) | [.(q−b)/(a−p),+infty.[ )
holds f **is** Lipschitzian;

theorem *:: FUZZY_6:29*
for a,b,p,q **being** Real, f **be** Function **of** REAL,REAL **st** a <> p **&**
f = (AffineMap (a,b) |].−infty,(q−b)/(a−p).[ )
+* (AffineMap (p,q) | [.(q−b)/(a−p),+infty.[ )
holds
f is_integrable_on A **&** f | A **is** bounded;

theorem *:: FUZZY_6:30*
for a,b,p,q **being** Real **st** a <> p **holds**
(AffineMap (a,b)) . ((q−b)/(a−p)) = (AffineMap (p,q)) . ((q−b)/(a−p));

theorem *:: FUZZY_6:31*
for f **be** Membership_Func **of** REAL **holds** f **is** bounded;

theorem *:: FUZZY_6:32*
for r **be** Real, f **be** Function **of** REAL,REAL **st**
r <> 0 **&** f is_integrable_on A **&** f | A **is** bounded

```
holds
centroid((r (#) f),A) = centroid(f,A);

theorem :: FUZZY_6:33
for a,b,c being Real, f,g,h be Function of REAL,REAL st
a <= b & b <= c &
f is_integrable_on ['a,c'] & f | ['a,c'] is bounded &
g is_integrable_on ['a,c'] & g | ['a,c'] is bounded &
h | [.a,c.] = (f| [.a,b.]) +* (g | [.b,c.]) &
h is_integrable_on ['a,c'] & f.b = g.b
holds
integral(h,['a,c']) = integral(f,['a,b']) + integral(g,['b,c']);

theorem :: FUZZY_6:34
for a,b,c be Real, f,g,h be Function of REAL,REAL st
a <= b & b <= c & f is continuous & g is continuous &
h = (f| [.a,b.]) +* (g | [.b,c.]) & f.b = g.b
holds
integral((id REAL) (#) h,['a,c'])
= integral((id REAL) (#) f,['a,b']) + integral((id REAL) (#) g,['b,c']);

theorem :: FUZZY_6:35
for c being Real, f,g be Function of REAL,REAL
holds
f|].-infty,c.[ +* g|[.c,+infty.[ = (f|].-infty,c.]) +* (g|[.c,+infty.[);

theorem :: FUZZY_6:36
for c being Real, f,g be Function of REAL,REAL st
f | A is bounded & g | A is bounded
holds
( (f|].-infty,c.]) +* (g|[.c,+infty.[) ) | A is bounded;

theorem :: FUZZY_6:37
for a,b,c being Real, f,g be Function of REAL,REAL st a <= c & c <= b
holds
f | [.a,c.[ +* g | [.c,b.] = f | [.a,c.] +* g | [.c,b.];

theorem :: FUZZY_6:38
for a,b,c being Real, f,g,h be Function of REAL,REAL st
a <= c & h | [.a,c.] = (f| [.a,b.]) +* (g | [.b,c.]) & f.b = g.b
holds
( b <= a implies h | [.a,c.] = g | [.a,c.] ) &
( c <= b implies h | [.a,c.] = f | [.a,c.] );

theorem :: FUZZY_6:39
for b being Real, f,g,h be Function of REAL,REAL st
h = (f| ].-infty,b.[) +* (g | [.b,+infty.[) & f.b = g.b holds
( b <= lower_bound A implies h | A = g | A ) &
( upper_bound A <= b implies h | A = f | A );

theorem :: FUZZY_6:40
for a,b,p,q being Real, f be Function of REAL,REAL st a <> p &
f = ( AffineMap (a,b) |].-infty,(q-b)/(a-p).[ )
 +* ( AffineMap (p,q) |[.(q-b)/(a-p),+infty.[ ) & (q-b)/(a-p) in A
holds
f | A = ( AffineMap (a,b) |[.lower_bound A,(q-b)/(a-p).] )
 +* ( AffineMap (p,q) |[.(q-b)/(a-p),upper_bound A.] );

theorem :: FUZZY_6:41
for a,b being Real holds
AffineMap (a,b) | A is bounded & AffineMap (a,b) is_integrable_on A;

theorem :: FUZZY_6:42
for a,b,p,q being Real, f be Function of REAL,REAL st a <> p &
f = ( AffineMap (a,b) |].-infty,(q-b)/(a-p).[ )
 +* ( AffineMap (p,q) |[.(q-b)/(a-p),+infty.[ )
holds
( (q-b)/(a-p) in A implies
```

```
integral(f,A) = integral(AffineMap (a,b),['lower_bound A,(q-b)/(a-p)'])
                 + integral(AffineMap (p,q),['(q-b)/(a-p),upper_bound A']) )
&
( (q-b)/(a-p) <= lower_bound A implies
integral(f,A) = integral(AffineMap (p,q),A) )
&
( (q-b)/(a-p) >= upper_bound A implies
integral(f,A) = integral(AffineMap (a,b),A) );

theorem :: FUZZY_6:43
for a,b,p,q being Real, f be Function of REAL,REAL st a <> p &
f | A = (AffineMap (a,b) | [.lower_bound A,(q-b)/(a-p).])
      +* (AffineMap (p,q) | [.(q-b)/(a-p),upper_bound A.]) & (q-b)/(a-p) in A
holds
integral((id REAL) (#) f,A)
= integral((id REAL) (#) (AffineMap (a,b)),['lower_bound A,(q-b)/(a-p)'])
+ integral((id REAL) (#) (AffineMap (p,q)),['(q-b)/(a-p),upper_bound A']);

theorem :: FUZZY_6:44
for a,b being Real holds
(id REAL) (#) (AffineMap (a,b)) = (a (#) ( #Z 2)) + (b (#) ( #Z 1));

theorem :: FUZZY_6:45
for a,b,c,d being Real st c <= d holds
integral((id REAL) (#) (AffineMap (a,b)),c,d)
= 1/3*a*(d*d*d - c*c*c) + 1/2*b*(d*d - c*c);

theorem :: FUZZY_6:46
for a,b being Real holds
AffineMap (a,b) = (a (#) ( #Z 1)) + (b (#) ( #Z 0));

theorem :: FUZZY_6:47
for a,b,c,d being Real st c <= d holds
integral(AffineMap (a,b),c,d) = 1/2*a*(d*d - c*c) + b*(d - c);

theorem :: FUZZY_6:48
for a,b,p,q,c,d,e being Real, f be Function of REAL,REAL st a <> p &
f | A = AffineMap (a,b) | [.lower_bound A,(q-b)/(a-p).]
      +* AffineMap (p,q) | [.(q-b)/(a-p),upper_bound A.] & (q-b)/(a-p) in A
holds
centroid(f,A) =
( 1/3*a*(((q-b)/(a-p))^3 - (lower_bound A)^3)
  + 1/2*b*(((q-b)/(a-p))^2 - (lower_bound A)^2)
  + 1/3*p*((upper_bound A)^3 - ((q-b)/(a-p))^3)
  + 1/2*q*((upper_bound A)^2 - ((q-b)/(a-p))^2) ) /
( 1/2*a*(((q-b)/(a-p))^2 -(lower_bound A)^2) + b*((q-b)/(a-p) - lower_bound A)
+ 1/2*p*((upper_bound A)^2 -((q-b)/(a-p))^2) + q*(upper_bound A -(q-b)/(a-p)) )
;

theorem :: FUZZY_6:49
for f being Function of REAL,REAL holds
max+ f = max(AffineMap(0,0),f);
```