

再帰的データ構造に基づく

2T-4

多次元ファジイ集合処理システムF++の開発

川 場 隆
活水女子短期大学

1. はじめに

計算機上で連続なファジイ集合を保持するために、少数の離散点データ(定義点座標とグレードの組み)を与えて、間を直線補完する方法がある。この方法は、メンバーシップ関数の変形が自由で演算速度も比較的速いという利点があるが、定義域が多次元になると計算効率の良い一般的な拡張が難しいという欠点がある。

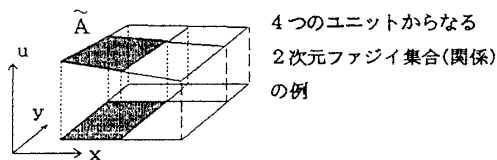
作成したファジイ集合処理系F++では、ユニットと呼ぶ再帰的なデータ構造を用いて、離散点データによる方法を多次元へ拡張した。C++のクラスライブラリとして実現しており、ファジイ集合、ファジイ数、タイプ2ファジイ数などを扱うことができる。

2. ユニットによるファジイ集合の表現

連続なファジイ集合 $\tilde{A}$ が、補完計算を前提として、 $X_1 \times X_2 \times \dots \times X_n$ 上の離散的な定義点(座標点とグレードの組み)で与えられているとき、その定義座標は、 X_i 上の離散的な要素集合、 $S_i = \{x_{i1}, x_{i2}, \dots, x_{im_i}\}$, $x_{i1} < x_{i2} < \dots < x_{im_i}$, $i=1, 2, \dots, n$ の直積集合として与えられる。

この時、 S_i の隣り合った二つの要素からなる部分集合 $S_{ik_j} = \{x_{ik_j}, x_{ik_{j+1}}\}$ の直積、 $S_{1k_1} \times S_{2k_2} \times \dots \times S_{nk_n}$ を定義座標とする、 $\tilde{A}$ の部分集合を、ユニットと呼ぶことにする。ユニットは、 2^n 個の、定義点座標とグレードの組みで表される。

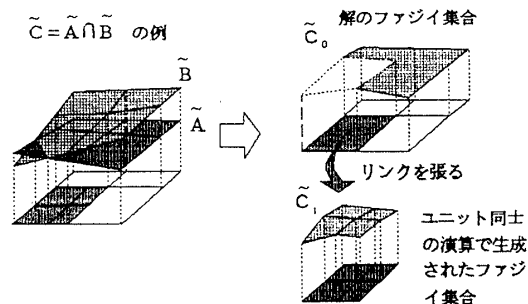
図1



F++ではファジイ集合をこのようなユニットの集

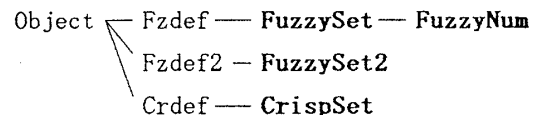
合体として扱い、集合演算はユニットを単位として実行する。演算結果もユニットとして得、最後に、それらをまとめて解のファジイ集合とする。ただし、互いに交差するユニット同士の演算では、ユニットはさらに細かく分割され、複数のユニットの集合体、すなわちファジイ集合が得られる。そこでこの場合は、元の座標点情報だけを持つ親ユニットを生成し、ここに、ユニット同士の演算で生成されたファジイ集合へのポインタを記録してリンクを張る。したがって、実際の演算では、ユニットがファジイ集合へのリンクポインタを持つかどうか判定し、持つ場合には集合演算を再帰的に適用することになる。

図2



3. クラス定義

"def"付きの基本クラスは、メンバーオブジェクトとしてユニットの動的配列と定義座標系や定義域情報を持ち、主として、検査やメンバーオブジェクトの参照、特定の領域からのユニットの切り出しなど、プリミティブな基本メソッドを有する。



集合演算メソッドは、太字で示した派生クラスが有する。FuzzySetクラスはファジイ集合(関係)、FuzzyNumクラスはファジイ数、FuzzySet2クラスはタイプ2ファジイ集合、CrispSetクラスは、アルファカットなどで使うクリスプ集合のクラスである。この中で、FuzzyNumクラスはFuzzySetクラスから派生しているので、ファジイ集合として可能な集合演

算メソッドを全て継承している。

F++にはこの他にも、座標を扱うためのベクトルクラスや、式文字列を解釈するパーサクラス、ユニットやファジイ集合の動的配列クラスなどがある。

4. 集合の生成記述 - コンストラクター

ファジイ集合の生成には、自然な式表現が使える。式文字列は、1)のようにプログラム中に直接書き込んだり、あるいは、2)のようにテキストファイルから読み込むこともできる。3)のように式文字列を指定しなければ、空集合が生成される。また、多次元のファジイ集合は、4)のように式文字列の座標値をベクトルで記述する。さらに、タイプ2ファジイ集合は、5)のように式文字列のグレードとして、ファジイ集合の式文字列を括弧でくくって指定する。

- 1) FuzzySet a("0.2/1+1.0/2+1.0/3+0/4");
- 2) FuzzySet b("sample.dat", "large");
- 3) FuzzySet c;
- 4) FuzzySet d("0.2/(1,1)+0.3/(1.2)+...");
- 5) FuzzySet2 e("(1/0+0.4/0.3+0/1)/3+...");

なお、三角型と台形型ファジイ数に限って、以上の方法に加えて、定義座標のみ指定する簡易生成を使うことができる。

- 6) FuzzyNum X(7, 10, 13); ← $0/7+1/10+0/13$
- 7) FuzzyNum y(4, 7, 8, 12); ← $0/4+1/7+1/8+0/12$

5. 演算定義と演算子、型変換

代入[=]、比較[==, !=, <=, >=]、アルファカット、分解定理、和(|, ||)、積(&, &&)、補(~)、射影[|], 補影[&], 円筒的拡張、含意[~], 合成[%]、非ファジイ化[!], 出力[>>]などが定義されており、[]内はその演算子である。演算子がないものはメンバー関数で記述するが、多くは、演算子をオーバーロードしてあり自然な形式で記述できる。また、メンバー関数名や演算子は、一部を除いてタイプ2ファジイ集合でも同じものが適用できる。

ファジイ数では、ファジイ数同士、あるいはファジイ数と実数との間で、加減乗除[+, -, *, /]とmax, minが定義されている。さらに、ファジイ数はファジイ集合から派生しているので、ファジイ集合として可能な演算は全て適用できる。

また、クリスプ集合では、代入、比較、和、積、補、出力などが定義されており、演算子はファジイ

集合に準じる。

なお型変換の点からは、ファジイ集合型はファジイ数の条件を満たしていれば、ファジイ数として扱って構わない。一方、ファジイ数型は無条件にファジイ集合型と混在して集合演算を適用できる。

例: FuzzySet x("0/3+1/4+0/6"), y;

```
FuzzyNum a(3, 5, 6), b;
b = x + a;
y = x & b;
```

6. 演算メソッドの変更

F++では、クラスの静的メンバー関数により、集合演算の基礎をなす和・積演算をはじめ、含意規則、合成演算の方法など、細目に渡る、メソッドの変更が可能である。変更は即時に有効となり、次に異なる変更が実行されるまで持続する。

- 1) FuzzySet::setTconorm(BND_S);
- 2) FuzzySet::setImplication(R2);
- 3) FuzzySet::setCompL(MIN);
- 4) FuzzySet::setCompR(BND_P);

1)は、和演算を限界和（デフォルトは論理和）に変更し、2)は含意規則をルカシェヴィッツの含意規則に（デフォルトはマムダニ）に変更する例である。

3), 4)は合成演算の変更例である。デフォルトの合成演算は、max-min合成を行うが、3)は、これをmin-min合成に変更し、4)は、max-限界積合成に変更する。これらの変更の選択肢として一般的なものはあらかじめ用意されているが、引数は関数ポインタであるから特殊な自作関数でも構わない。

7. おわりに

F++は、C++の動作する全ての環境で動作する。ただ、現状では離散的ファジイ集合を扱えないことやデータの出力形式が見つらいなどの欠点がある。今後これらの諸点に留意し、汎用システムとしてさらに拡張を行っていきたい。

参考文献:

- 1) 馬野、鳩野、田村「コンピューターによるファジイ集合の表現と処理」第9回ファジイシステムシンポジウム, pp77-80
- 2) 藤井、茂木、Harper、中嶋、十河、荒尾:「実用化を目指したファジイC言語の開発」第9回ファジイシステムシンポジウム, pp533-536
- 3) 日本ファジイ学会:「ファジイ集合」, 日刊工業新聞社, 1992