



**HAL**  
open science

## Graphes de Voronoï Flous basés sur un nouvel opérateur de Tri des Nombres Flous Triangulaires

Aziz Mabrouk, Azedine Boulmakoul, Mohamed Haitam Laarabi, Roberto  
Sacile, Emmanuel Garbolino

► **To cite this version:**

Aziz Mabrouk, Azedine Boulmakoul, Mohamed Haitam Laarabi, Roberto Sacile, Emmanuel Garbolino. Graphes de Voronoï Flous basés sur un nouvel opérateur de Tri des Nombres Flous Triangulaires. 3ème Edition du workshop "Innovation et Nouvelles Tendances dans les Systèmes d'Information" INTIS'2013, ENSA Tanger, Nov 2013, Tanger, Maroc. hal-01081948

**HAL Id: hal-01081948**

**<https://minesparis-psl.hal.science/hal-01081948>**

Submitted on 9 Dec 2014

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Graphes de Voronoï Flous basés sur un nouvel opérateur de Tri des Nombres Flous Triangulaires

Aziz Mabrouk\*, Azedine Boulmakoul\*\*

Mohamed Haitam Laarabi\*\*\*, Roberto Sacile\*\*\*, Emmanuel Garbolino\*\*\*\*

\* Faculté Polydisciplinaire de Tétouan

Département de Mathématiques et Informatiques, Route Martil -Tétouan Maroc

aziz.mabrouk@gmail.com

\*\* LIM/IST Lab., Computer Sciences Department, Faculté des sciences et technique de Mohammedia (FSTM), B.P. 146 Mohammedia, Morocco.

azedine.boulmakoul@yahoo.fr

\*\*\* Department of Computer science, Bioengineering, Robotics and Systems engineering (DIBRIS), University of Genoa, 16145 Genoa GE, Italy

haitam.laarabi@unige.it, roberto.sacile@unige.it

\*\*\*\* Crisis Research Centre (CRC), Mines ParisTech, Sophia Antipolis, France

emmanuel.garbolino@mines-paristech.fr

**Résumé.** Le tri des nombres flous constitue une opération principale pour calculer les graphes de Voronoï flous. Dans ce papier, nous proposons un nouvel opérateur de tri. C'est une fonction, qui malgré sa limite aux nombres flous triangulaires et trapézoïdaux, elle repose sur un minimum d'opérations mathématique et par conséquent un temps réduit pour construire les graphes de Voronoï flous. En effet, nous intégrons ce nouvel opérateur de tri dans notre processus de calcul des graphes de Voronoï flous que nous présentons également dans ce papier. Enfin, nous présentons le nouveau composant logiciel développé pour construire ces structures géométriques.

## 1 Introduction

Les diagrammes de Voronoï de type réseau classique sont discutés par (Okabe et al., 2002a) et également dans une revue concernant les problèmes d'optimisation de localisation (Okabe et Suzuki, 1997). Ainsi, plusieurs algorithmes ont été proposés dans la littérature (Erwig, M., 2000), (Takehiro et al., 2005), (Margot Graf et Stephan Winter, 2003) et qui permettent de calculer le diagramme de Voronoï de type réseau dont le poids des chemins parcourus est considéré une valeur réel. Or, cette estimation ne reflète pas parfaitement la réalité (Mabrouk et Boulmakoul, 2013). En effet, nous avons prolongé ces algorithmes en se basant sur une modélisation floue d'un réseau réel (réseau routier, réseau de transport, etc...) par un graphe flou. Ceci implique la recherche des plus courts chemins flous pour mettre en relation les nœuds de ce graphe flou à leurs plus proches générateurs de Voronoï (Ecoles, hôpitaux, etc.).

Dans de nombreuses applications de la théorie des ensembles flous à la prise de décision, nous sommes confrontés au problème de sélection d'une solution parmi un ensemble de solutions possibles, et en général, nous voulons connaître la meilleure. Ainsi au niveau des

problèmes de recherches du plus court chemin, nous avons besoins de sélectionner un chemin parmi d'autres avec des coûts à valeurs flou. Cela revient à trier des valeurs floues de manière approximative et c'est toujours un sujet de controverse. En effet, cela n'est pas aussi simple que le tri d'ensembles nets (tel que IR), due à la linéarité de ces ensembles. Les opérateurs  $<$ ,  $\leq$  et  $=$  ne peuvent pas être directement étendue aux ensembles flous (Zadeh, 1965).

Par conséquent, plusieurs propositions ont émergé abordant cette problématique. (Yao et Wu, 2000) ont proposé une méthode de classement avec distance signée, ce qui a permis l'évaluation de la distance entre deux nombres flous. (Yager, 1981) ont utilisé un nombre flou à valeur pondérée. (Wang et al., 2006) ont proposé une méthode de calcul de distance entre barycentre. (Lee-Kwang et Lee, 1999) ont introduit l'évaluation, basé sur le point de vue de l'utilisateur, des ensembles flous comme une pré-étape pour classement en utilisant une fonction de satisfaction. Il existe bien plus autres méthodes comme dans (Chou et al, 2011), (Ezzati et al., 2012), (Chen, 1985), (Abbasbandy et Hajjari, 2009), (Chu et Tsao, 2002), (Wang et Lee, 2008), (Cheng, 1998), (Choobineh et Li, 1993) et (Baldwin, 1979).

En fait, le temps de construction du Graphe de Voronoï Flou est calculé principalement en fonction de nombre des nœuds et des arcs constituant le graphe flou. Par conséquence, ce processus traite une grande quantité de données notamment dans le cas d'un réseau spatial réel. En effet, ces méthodes de trie peuvent ne pas être suffisamment efficaces lors du traitement de cette masse de données, puisqu'elles reposent sur beaucoup d'opérations de calcul. Ainsi, nous proposons une méthode de classement qui assure la réduction des opérations et des étapes, au point de prendre directement des décisions sans opérations de comparaison. Afin de simplifier l'explication de notre approche, nous allons l'appliqué sur un cas particulier des ensembles flous : les Nombres Flous Triangulaires (NFT). Quoique ça peut être facilement étendu au Nombres Flous Trapézoïdal.

L'opérateur de tri proposé profite de l'encodage structurel relatif à la relation topologique entre deux triangles. En outre, l'indice d'inclusion – un indice indiquant le degré d'inclusion dans le minimum de deux nombres flous – est considéré comme un indicateur approchant le « Fortement inclus dans ».

## 2 Fonctions de Tri des nombres flous

### 2.1 Fonctions de tri existantes

#### 2.1.1 Opérateur de treillis MIN et MAX

La méthode proposée pour classement des NFTs est basée principalement sur les opérateurs de treillis *MIN* et *MAX*. En effet, (Dubois et Prade, 1978) et (Klir et Yuan, 1995) ont veillé à ce que le triplet  $\langle R, MIN, MAX \rangle$  est un treillis distributif, dans lequel *MIN* et *MAX* représentent l'intersection et l'union, respectivement. Il est nécessaire de souligner donc que l'utilisation des opérateurs des nombres réels min et max ne sont pas applicables, car ils sont exprimés avec des termes du couple  $(R, \leq)$ . En étendant min et max sur NFTs, nous pouvons les utiliser pour formuler la méthode proposée. La définition de l'opérateur de treillis *MIN* est comme suite :

$$MIN(A, B)(z) = \sup_{z=\min(x,y)} \min[A(x), B(y)] \quad (1)$$

Pour tout NFTs  $A$  et  $B$ , et pour tout  $x, y, z \in R$ . (Chiu et Wang, 2002) introduisent un théorème simplifiant l'implémentation des opérateurs  $MIN$  and  $MAX$ , comme suite :

Théorème 1 (Chiu Wang, 2002). Pour tout NFTs  $A$  et  $B$ , définis sur l'ensemble universel  $R$ , avec une fonction d'appartenance continue et  $(A \cap B) \neq \emptyset$ , soit  $x_m \in R$  un point tel que  $(A \cap B)(x_m) \geq (A \cap B)(x)$  pour tout  $x \in R$  et  $A(x_m) = B(x_m)$ , en outre,  $x_m$  est entre deux valeurs moyennes de  $A$  et  $B$  (si le nombre  $x_m$  n'est pas unique, alors n'importe quel  $x_m$  est convenable). L'opérateur  $MIN$  peut être implémente comme suite :

$$MIN(A, B)(z) = \begin{cases} (A \cup B)(z) & \text{asz} < x_m \\ (A \cap B)(z) & \text{asz} \geq x_m \end{cases} \quad (2)$$

Sachant que  $z \in R$  et  $\cup$  et  $\cap$  indiquent respectivement la standard intersection et union flou.

Le théorème ci-dessus fournit une simple procédure pour la mise en œuvre de l'opérateur  $MIN$ . Il facilite le contrôle rapide des résultats promis par l'opérateur de tri implémenté.

### 2.1.2 L'index d'inclusion

C'est un indicateur quantitatif exprimant le degré d'inclusion, dont définition consiste à considérer que  $E \subseteq F \Leftrightarrow (card(E \cap F) = card(E))$  avec  $E$  et  $F$  deux ensembles flous. Alors le degré d'inclusion  $\partial(E \subseteq F)$ , ou  $card_E(E \subseteq F)$ , est donnée par :

$$\partial(E \subseteq F) = \frac{\frac{\sum |E \cap F|}{\sum |E|}}{\sum_{x \in X} T(\mu_E(x), \mu_F(x))} \quad (3)$$

Avec  $T$  est la norme triangulaire et  $||$  dénote le standard opérateur flou de cardinal  $card$ .  $\mu_E$  et  $\mu_F$  est respectivement la fonction d'appartenance de  $E$  et  $F$ .

## 2.2 L'opérateur de tri proposé

Pour tous les ensembles flous  $A$  et  $B$ , les opérateurs de classement  $<, >$  et  $\simeq$  sont définis par les implications suivantes :

$$\begin{array}{l} \text{If } MIN \in \{A, B\} \\ \left\{ \begin{array}{l} A < B \Leftrightarrow MIN = A \\ A > B \Leftrightarrow MIN = B \\ A \simeq B \Leftrightarrow MIN = A \text{ et } MIN = B \end{array} \right. \end{array} \quad \begin{array}{l} \text{Else} \\ \left\{ \begin{array}{l} A < B \Leftrightarrow \partial(MIN \subseteq A) > \partial(MIN \subseteq B) \\ A > B \Leftrightarrow \partial(MIN \subseteq A) < \partial(MIN \subseteq B) \\ A \simeq B \Leftrightarrow \partial(MIN \subseteq A) = \partial(MIN \subseteq B) \end{array} \right. \end{array} \quad (4)$$

La majorité des cas relatifs aux relations topologique entre deux NFTs permettent de déduire directement l'index d'inclusion sans aucun calcul a appliqué. Ainsi, quatre principales situations topologiques sont possibles entre deux NFTs (On suppose que  $A$  et  $B$  représentent deux  $TfNs$ , avec  $A^- < B^-$ ) :

- Disjoint : Les deux NFTs ne se chevauchent pas, avec  $A^+ < B^-$  et visiblement  $MIN(A, B) = A$ , donc  $A < B$  ;

## Graphes de Voronoï Flous basés sur un opérateur de Tri des Nombres Flous Triangulaires

- Chevauchement faible : Les deux NFTs se chevauchent faiblement tel que, avec  $A^0 < B^-$  et visiblement  $MIN(A, B) = A$ , donc  $A < B$  ;
- Chevauchement : Nous avons  $A^+ < B^+$ , et donc tous les cas on obtient  $A < B$ , sauf pour le cas ou  $B^0 < A^0$  ou il faut calculer la surface des triangles.
- Inclusion : Représente tous les cas ou  $A^- < B^-$  et  $A^+ > B^+$ . Nous obtenons  $A < B$  quand  $A^0 < B^0$ , et le calcul de la surface devient nécessaire pour les cas contraires.
- Comme résultat, notre approche nous a permis de déduire directement dans 70% des cas, la comparaison de deux NFTs.

### 3 Graphe de Voronoï Flou

#### 3.1 Définition

Nous définissons le Graphe de Voronoï Flou (GV-Flou) par la division d'un graphe flou en sous-graphe flou dont chacun contient les nœuds et les arcs les plus proches à chaque générateur de Voronoï en parcourant les plus courts chemins flous pour mettre en relation ces composantes géométriques (Mabrouk et al., 2009).

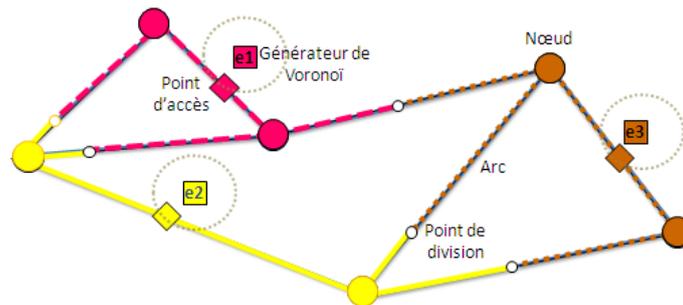


FIG. 1 – Graphe de Voronoï Flou généré par les générateurs de Voronoï  $e1$ ,  $e2$  et  $e3$

Soit  $N(S, A)$  un graphe flou de sommets  $S$  et d'arcs  $A$  et  $P$  un ensemble de sommets  $G = \{g_1, \dots, g_n\}$  avec  $G \subseteq S$ .  $G$  représentent les générateurs de Voronoï dans le GV-Flou. Chaque poids valant un arc, est représenté par un nombre flou. Considérons  $v$  et  $w$  deux sommets appartiennent à  $S$ . Nous allons utiliser  $P_{flou}(v, w)$  pour représenter le poids flou du plus court chemin de  $v$  à  $w$  dans le réseau  $N$ . Le GV-Flou pour  $G$  divise le graphe  $N$  en  $n$  sous graphe flou  $Vor_{flou}(1), \dots, Vor_{flou}(n)$  avec :

$$Vor_{flou}(i) = \{\forall p \in P / P_{flou}(pi, p) \leq P_{flou}(pj, p), 1 \leq \forall j \leq n, i \neq j\} \quad (5)$$

#### 3.2 Notre processus de calcul du GV-Flou

L'algorithme que nous présentons dans ce papier a été publiée également dans (Mabrouk et al., 2009) et (Mabrouk et Boulmakoul, 2008c). En effet, le pseudo code de cette méthode est présenté comme le suivant :

---

```

Initialiser :
  Pour chaque  $v \in S$  faire
    Si  $v \in G_n$  alors //  $v$  est un générateur
       $Gt := v$  ;  $Ppcc_{f1ou}(v) \equiv (0, 0, 0)$  ;  $Gp(v) := \Gamma$  ; insérer( $v, Q$ ) ;
    Sinon
       $Gt(v) := \Gamma$  ;  $Ppcc_{f1ou}(v) \equiv (\infty, \infty, \infty)$  ;  $Gp(v) := \Gamma$  ;
    Fin Si
  Fin Pour
Fin Initialiser
Marquer :
   $v := \text{Extraitre\_Min}(Q)$  ;  $Gp(v) := Gt(v)$  ; marquer  $v$  ;
Fin Marquer
Balayer :
  Soit  $\text{Adj}[v]$  l'ensemble des sommets voisins de  $v$  sélectionné.
  Pour chaque  $w \in \text{Adj}[v]$  n'est pas marqué faire
     $\Delta_{f1ou} \equiv Ppcc_{f1ou}(w) \oplus P_{f1ou}(v, w)$ 
    Si  $Ppcc_{f1ou}(w) \equiv (\infty, \infty, \infty)$  alors
       $Gt(w) := Gt(v)$  ;  $Ppcc_{f1ou}(w) \equiv \Delta_{f1ou}$  ; insérer( $w, Q$ ) ;
    Fin Si

    Si  $R(Ppcc_{f1ou}(w), \Delta_{f1ou}) \geq 0$  alors
       $Gt(w) := Gt(v)$  ;  $Ppcc_{f1ou}(w) \equiv \Delta_{f1ou}$  ;
      Mettre_à_jour( $Q, w, \Delta_{f1ou}$ ) ;
    Fin Si
  Fin Pour
Fin Balayer
Contrôler
  Tant que  $Q$  n'est vide faire
    Marquer et balayer
  Fin Tant que
Fin Contrôler

```

---

$Ppcc_{f1ou}(v)$  dénote le poids flou du plus court chemin, du plus proche générateur de Voronoï temporaire  $Gt(v)$  au nœud  $v$ .

$R()$  est une fonction qui utilise l'opérateur de tri proposé dans ce papier pour comparer deux nombres flous  $A(A^-, A^0, A^+)$  et  $B(B^-, B^0, B^+)$  et ce, en se basant sur les valeurs de  $A^-, A^0, A^+, B^-, B^0$  et  $B^+$  (voir la section 2.2) dont :

$$A \geq B \text{ si et seulement si } R(A, B) \geq 0 \quad (6)$$

$\oplus$  dénote l'addition des nombres flous.

En considérant les générateurs de Voronoï comme des sources multiples, notre processus cherche en parallèle les nœuds les plus proches à chaque générateur en se basant sur la comparaison des poids flous des chemins parcourus.

Nous proposons l'utilisation d'une affectation floue " $\equiv$ " dans la première étape pour initialiser les poids des plus courts chemins, parcourus pour atteindre les générateurs de Voronoï.

noï et le reste des nœuds, respectivement par des nombres flous nuls  $Ppcc_{flou}(v) \equiv (0,0,0)$  et des nombres flous infinis  $Ppcc_{flou}(v) \equiv (\infty, \infty, \infty)$ .

Dans l'étape « Balayer », nous utilisons l'affectation floue "≡" et l'addition floue  $\oplus$  pour l'addition des nombres flous. Egalement, nous utilisons la fonction de classement  $R(\ )$  et de comparaison "≡" pour comparer les valeurs floues des poids des chemins parcourus.

## 4 Composants logiciels calculant le Graphe de Voronoï Flou

### 4.1 Description des composants logiciels développés

Vu que la technologie COM a été développée pour permettre la communication interprocessus, nous avons développés des composants logiciels avec Visual C++ en utilisant la bibliothèque de modèles actifs ATL (Active Template Library). Ce sont des objets COM (Component Object Model) indépendants (sous forme de dynamic link library DLL) qui calculent le Graphe de Voronoï Flou.

Ils reposent sur un système de communication via interface composé en fait d'une série de fonctions accessible aux autres programmes. En effet, ces composants sont également réutilisables par des applications SIG à savoir le logiciel SIG ESRI® ArcMap 9.2.

Le processus de calcul du GV-Flou reçoit en entrée d'une part, le graphe flou et d'autre part, les générateurs de Voronoï. Considérant ces générateurs comme des sources multiples, il cherche en parallèle les plus courts chemins flous entre ces générateurs et le reste des nœuds du réseau spatial.

Chaque composant logiciel exécute trois processus principaux :

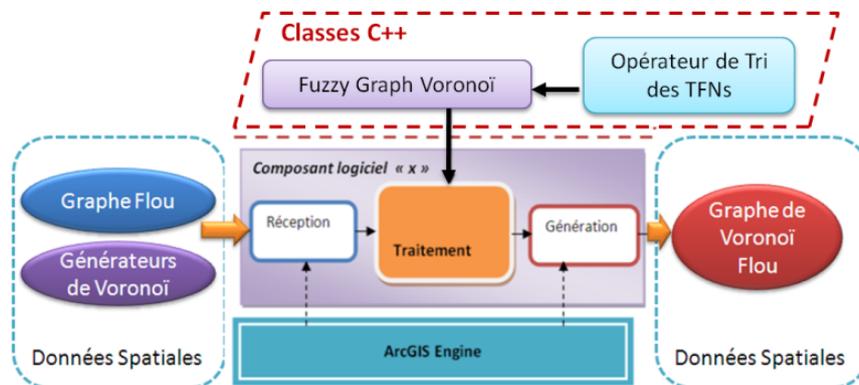


FIG. 2 – Architecture du composant logiciel calculant le Graphe de Voronoï Flou basé sur le nouvel opérateur de tri des NFTs.

#### 4.1.1 Processus de réception et génération des données spatiales

Ces deux processus sont assurés par ArcGIS Engine Runtime. Ceci constitue une bibliothèque permettant d'accéder et gérer des données spatiales. En utilisant les classes et les interfaces de cette bibliothèque, le composant logiciel accède aux propriétés géométriques et

alphanumériques des données spatiales stockées dans des fichiers de formes ou des classes d'entités d'une base de données géographique. En se basant sur ces données reçues et selon le traitement nécessaire à la réalisation de la fonction demandée, le composant logiciel construit des nouvelles données spatiales et les stocke ensuite dans des nouveaux fichiers de formes ou des classes d'entités d'une base de données géographique.

#### **4.1.2 Processus de traitement des données spatiales**

Ce processus reçoit en entrée les propriétés géométriques (forme, distance, surface, nombre d'éléments,...) et alphanumériques (attributs) des données spatiales reçues par le processus « réception ». Et en se basant sur les méthodes que nous avons proposées et sur les classes C++ que nous avons implémentées (notamment les classes calculant le Graphe de Voronoï Flou en se basant sur l'opérateur de tri proposé dans ce papier), le processus met en sortie les éléments nécessaires pour construire ou mettre à jour des données spatiales par le processus « génération ».

### **4.2 Composant logiciel construisant les nœuds du GV-Flou**

Ce composant logiciel reçoit en entrée le fichier de formes comportant les nœuds du réseau y compris les générateurs de Voronoï, le fichier de formes des générateurs de Voronoï (Ecoles, hôpitaux, etc. ...) et le fichier de formes des arcs comportant les trois champs stockant les bornes des fonctions d'appartenance qui sont relatives au poids flou de chaque arc du graphe flou (Mabrouk et Boulmakoul, 2010).

En utilisant l'algorithme que nous proposons en haut, ainsi publié dans (Mabrouk et al., 2009), le composant logiciel construit le graphe flou et puis procède parallèlement à la recherche des nœuds qui sont plus proche à chaque générateur de Voronoï en parcourant les plus courts chemins.

Il génère alors en sortie un fichier de formes semblable au fichier de formes des nœuds mais avec quatre champs supplémentaires pour stocker pour chaque nœud le générateur associé et le total des poids flous (trois valeurs réelles) pour atteindre ce générateur.

### **4.3 Composant logiciel pour l'affectation des arcs d'un graphe flou à leurs plus proches générateurs de Voronoï**

Il reçoit en entrée le fichier de forme des arcs et le fichier de forme des nœuds généré par le composant logiciel présenté dans la section précédente (les nœuds du GV-Flou). Ce fichier des nœuds comporte des champs contenant des données sur les générateurs de Voronoï associées aux nœuds et les poids flous des plus courts chemins parcourus pour atteindre ces générateurs.

D'une manière similaire au composant logiciel affectant les arcs d'un réseau crisp à leurs plus proches générateurs de Voronoï (Mabrouk et Boulmakoul, 2008a), Ce composant logiciel marque chaque arc du réseau spatial selon l'affectation de son nœud de début et celle de fin aux générateurs de Voronoï.

Si le nœud de début et celui de la fin appartiennent aux différents générateurs, et l'arc déterminé par ces deux nœuds n'est pas nécessairement symétrique et il est accessible par n'importe quel point qui lui appartient. Dans ce cas l'arc doit être divisé. Le point de division

est le point  $C(X_C, Y_C)$  auquel le poids flou de déplacement  $Poids_{flou}(C, g(s))$  au générateur  $g(s)$  associé au nœud de début "s", est égal au poids flou de déplacement  $Poids_{flou}(C, g(s'))$  au générateur  $g(s')$  associé au nœud de fin "s". Alors:

$$Poids_{flou}(C, g(s)) = Poids_{flou}(C, g(s')) \quad (7)$$

La première part  $[sC]$  (du nœud de début « s » au point de division « C ») sera assigné à l'arbre des plus courts chemins auquel appartient le nœud de début « s », et la deuxième part,  $[Cs']$  (du point de division « C » au nœud de fin « s' ») sera assignée à l'arbre des plus courts chemins auquel appartient le nœud de fin « s' ».

#### 4.4 Calcul des références spatiales floues d'un point de division

Soit  $A$  l'ensemble des arcs d'un réseau spatial flou. Soit  $Q \subset A$  l'ensemble des arcs dont les générateurs de leurs extrémités sont différents. Soit  $[NM] \in Q$  l'un de ces arcs et qui se compose d'un ensemble de segments. Soit  $[AB] \in [NM]$ , le segment où le point flou de division  $C(X_C, Y_C)$  coupe  $[NM]$  en deux parts :  $[NC]$  et  $[CM]$  Avec :

$$\begin{cases} NM = NC + CM \\ P_{flou}(N, g(N)) + NC = P_{flou}(M, g(M)) + CM \end{cases} \quad (8)$$

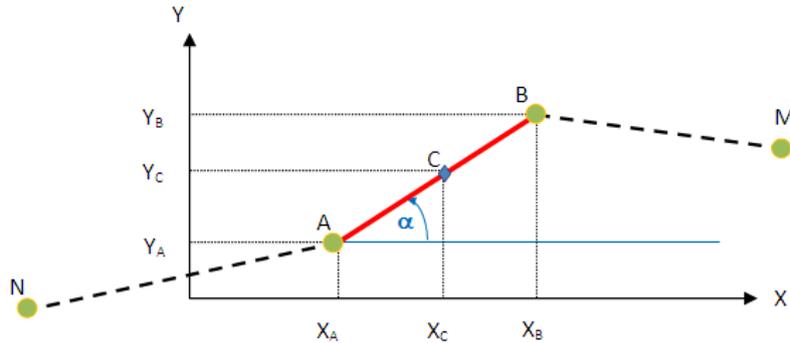


FIG. 3 – Le point de division  $C$  coupe le segment  $[AB]$  en 2 parts  $[AC]$  et  $[CB]$  et coupe l'arc  $[NM]$  en deux arcs  $[NC]$  et  $[CM]$

D'après (Mabrouk et Boulmakoul, 2013) :

$$X_C = \left( \frac{Poids_{flou}(M, g(M)) - Poids_{flou}(N, g(N)) + NM}{2} - NA \right) \left( \frac{X_B - X_A}{AB} \right) + X_A \quad (9)$$

$$Y_C = \left( \frac{Poids_{flou}(M, g(M)) - Poids_{flou}(N, g(N)) + NM}{2} - NA \right) \left( \frac{Y_B - Y_A}{AB} \right) + Y_A \quad (10)$$

$X_C$  et  $Y_C$  sont alors des sous-ensembles flous déterminés respectivement par les bornes  $(x_{CL}, x_C, x_{CR})$  et  $(y_{CL}, y_C, y_{CR})$  avec  $x_{CL}, x_C, x_{CR}, y_{CL}, y_C$  et  $y_{CR}$  sont des nombres réels. Le point flou  $C(X_C, Y_C)$  est alors un ensemble de points  $c(x, y)$  avec  $x \in \mathbb{R}$  et  $y \in \mathbb{R}$

Soit  $\mu_{X_C}(x)$  et  $\mu_{Y_C}(y)$  respectivement les fonctions d'appartenance de  $X_C$  et  $Y_C$   
 $\forall \alpha \in [0,1]$ :

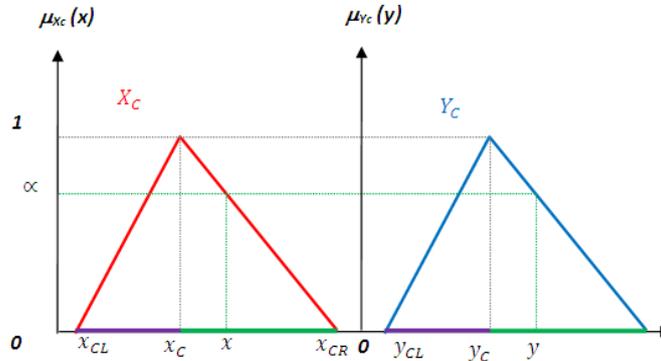


FIG. 4 – Les références spatiales floues d'un point de division

$$\text{Si } x_{CL} < x < x_C \text{ alors } \alpha = \mu_{X_C}(x) = (x - x_{CL}) / (x_C - x_{CL}) \\ \Rightarrow x = \alpha \cdot (x_C - x_{CL}) + x_{CL}$$

$$\text{Si } x_C < x < x_{CR} \text{ alors } \alpha = \mu_{X_C}(x) = (x_{CR} - x) / (x_{CR} - x_C) \\ \Rightarrow x = x_{CR} - \alpha \cdot (x_{CR} - x_C)$$

D'une manière similaire,

$$\text{Si } y_{CL} < y < y_C \text{ alors } \alpha = \mu_{Y_C}(y) = (y - y_{CL}) / (y_C - y_{CL}) \\ \Rightarrow y = \alpha \cdot (y_C - y_{CL}) + y_{CL}$$

$$\text{Si } y_C < y < y_{CR} \text{ alors } \alpha = \mu_{Y_C}(y) = (y_{CR} - y) / (y_{CR} - y_C) \\ \Rightarrow y = y_{CR} - \alpha \cdot (y_{CR} - y_C)$$

Pour chaque  $0 \leq \alpha < 1$ , on distingue deux type de GV-Flou pour le même graphe flou et les mêmes générateurs de Voronoï (figure 5).

- GV-Flou Gauche calculé en fonction des valeurs appartenant au support de la partie gauche des poids flous des plus courts chemins et qui sont représentés par des NFTs;
- GV-Flou Droit calculé en fonction des valeurs appartenant au support de la partie droite des poids flous des plus courts chemins et qui sont représentés par des NFTs.

Pour  $\alpha = 1$ , le Graphe de Voronoï résultant est un Diagramme de Voronoï de type réseau classique (figure 5).

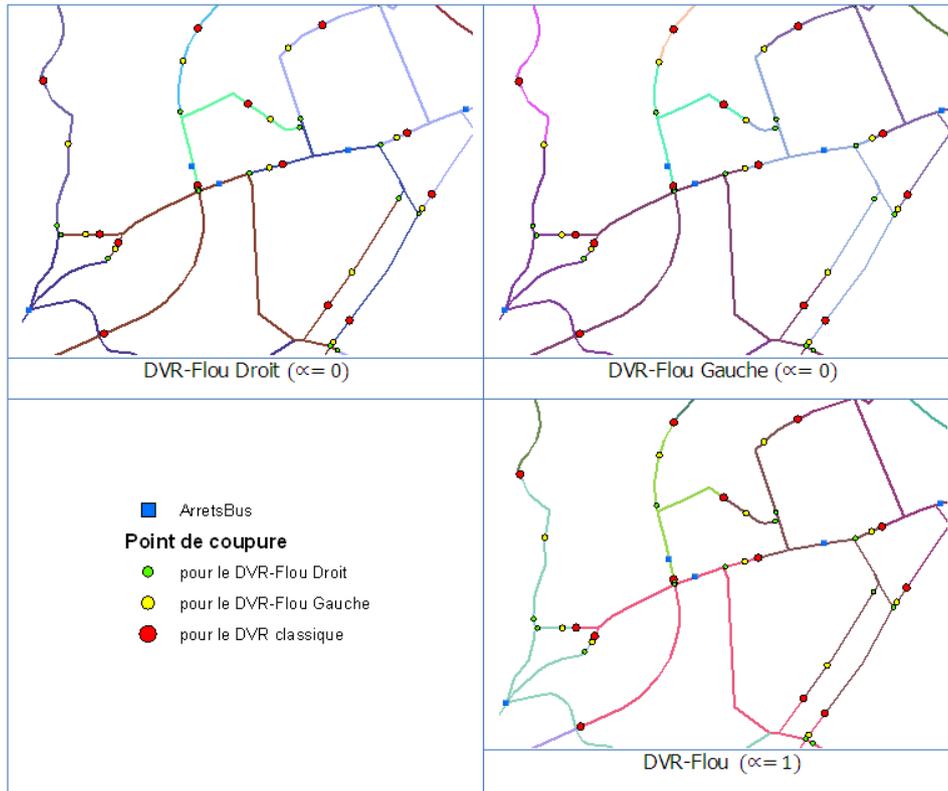


FIG. 5 – Extrait des GV-Flou selon le type et la valeur de  $\alpha$ , pour les arrêts de bus de la ville de Tétouan

## 5 Conclusion

En comparaison avec les fonctions de tri présentées dans la littérature, l'opérateur de tri des NFTs, proposé dans ce papier, a permis de réduire le temps de calcul du Graphe de Voronoï Flou. On constate que ceci est très bénéfique en termes de temps dans certains domaines d'application qui demandent des réponses et des décisions rapides. Néanmoins, cet opérateur reste limité seulement aux nombres triangulaires et trapézoïdaux. En effet, la perspective de ce travail, c'est de proposer des méthodes capables de trier tous les formes des nombres flous.

## Références

- Abbasbandy, S. et Hajjari, T. (2009). A new approach for ranking of trapezoidal fuzzy numbers. *Computers & Mathematics with Applications*, 57, 413-419
- Baldwin, J.F. et Guild, N.C.F. (1979). Comparison of fuzzy sets on the same decision space, *Fuzzy Sets Systems*, 2, pp. 213–233
- Chen, S. (1985). *Ranking fuzzy numbers with maximizing set and minimizing set*, *Fuzzy sets and Systems*, 17, 113-129.
- Cheng, C. (1998). *A new approach for ranking fuzzy numbers by distance method*, *Fuzzy sets and systems*, 95, 307-317.
- Chiu, C. et Wang, W. (2002). *A simple computation of min and max operations for fuzzy numbers*. *Fuzzy Sets and Systems*, 126, 273-276
- Chou, S., L. Dat, V. Yu. (2011). *A revised method for ranking fuzzy numbers using maximizing set and minimizing set*, *Computers & Industrial Engineering*, 61, 1342-1348.
- Choobineh, F et H. Li. (1993). *An index for ordering fuzzy numbers*, *Fuzzy Sets and Systems*, 54, 287-294.
- Chu, T. et Tsao, C. (2002), *Ranking fuzzy numbers with an area between the centroid point and original point*, *Computers & Mathematics with Applications*, 43, 111-117.
- Dubois, D. et Prade, H. (1978). *Operations on fuzzy numbers*. *International Journal of systems science*, 9, 613-626
- Erwig Martin et Hagen Fernuniversitat (2000). *The graph Voronoi diagram with applications*. *Networks*, Citeseer, Vol. 36, pp.156-163
- Ezzati R., T. Allahviranloo, S. Khezerloo et M. Khezerloo (2012). *An approach for ranking of fuzzy numbers*, *Expert Systems with Applications*, 39, 690-695.
- Klir, G. et Yuan, B. (1995). *Fuzzy sets and fuzzy logic: Theory and Applications*. Prentice Hall New Jersey
- Lee-Kwang, H. et Lee, J. (1999). *A method for ranking fuzzy numbers and its application to decision-making*, *Fuzzy Systems, IEEE Transactions on*, 7, 677-685.
- Mabrouk, A. et Boulmakoul A., (2013). *Infrastructure logicielle intégrant un système spatial décisionnel pour la géo-gouvernance des réseaux urbains*. ASD'13, Systèmes intelligents théories et applications, –Marrakech, Mai.
- Mabrouk, A. et Boulmakoul A., (2010). *Système spatial interactif d'aide à la décision basé sur les diagrammes de Voronoï flous*. la sixième conférence internationale SITA'10, Systèmes intelligents théories et applications, ENSIAS –Rabat, Mai.

- Mabrouk, A., Boulmakoul, A. et Bielli, M. (2009). *Fuzzy spatial network Voronoï diagram: a spatial decision support for transportation planning*, Int. J. Services Sciences, Vol. 2, Nos. 3/4, pp.265–280.
- Mabrouk, A., et Boulmakoul (2008a). Réseau Spatial Flou de Voronoï Un Support d’Aide à la Décision pour la planification urbaine, in : Les systèmes décisionnels applications et perspectives, Atelier sur les systèmes décisionnels, ASD 2008, 2168/2008,pp. 161-170, ISBN: 978-9981-1-3000-1
- Mabrouk, A. et Boulmakoul A., (2008c). *Processus de calcul du diagramme de Voronoï de type réseau basé sur la modélisation floue des réseaux spatiaux*. journées JOSTIC’08, Association ACTIF & le laboratoire LIMIARF, Faculté des sciences de Rabat, Novembre
- Margot Graf et Stephan Winter (2003). *Network voronoi diagramme*. Österreichische Zeitschrift für Vermessung und Geoinformation, Vol. 91, pp.166-174
- Okabe, A., Okunuki, K. et Funamoto, S. (2002a). SANET: A Toolbox for Spatial Analysis on a Network, Center for Spatial Information Science, University of Tokyo.
- Okabe, Atsuyuki and Suzuki, Atsuo, (1997). *Locational optimization problems solved through Voronoi diagrams*. European Journal of Operational Research, Elsevier, Vol. 98, pp.445-456
- Takehiro Furuta, Suzuki Atsuo et Inakawa Keisuke (2005). *The kth nearest network Voronoi diagram and its application to districting problem of ambulance systems*. Discussion Paper
- Wang, Y., J. Yang, D. Xu, K. Chin (2006). *On the centroids of fuzzy numbers*, Fuzzy Sets and Systems, 157, 919-926.
- Wang, Y. et H. Lee (2008). *The revised method of ranking fuzzy numbers with an area between the centroid and original points*, Computers & Mathematics with Applications, 55, 2033-2042.
- Yao, J. et Wu, K. (2000). *Ranking fuzzy numbers based on decomposition principle and signed distance*, Fuzzy Sets and Systems, 116, 275-288.
- Yager, R. (1981). *A procedure for ordering fuzzy subsets of the unit interval*, Information Sciences, 24, 143-161.
- Zadeh, L. A., (1965), *Fuzzy sets, Information and Control*, Vol. 8, pp.338–353.

## Summary

The ranking of fuzzy numbers is a main operation to calculate the Fuzzy Voronoi Graphs. In this paper, we propose a new rank operator. This is a function which, despite its limited to triangular and trapezoidal fuzzy numbers, it is based on a minimum of mathematical operations and therefore less time to build the Fuzzy Voronoi Graphs. In fact, we integrate this new ranking operator in our calculation process of fuzzy Voronoi Graphs we present also in this paper. Finally, we present the new software component developed to build these geometric structures.