Séminaire Mathématique de Béjaïa (LaMOS)
Volume 17, 2017, pp. 19 - 22

4

Clustering, Jeux et Contraintes

K. BOUCHAMA et M.S. RADJEF

Laboratoire de Modélisation et d'Optimisation des Systèmes (LAMOS)
Université de Béjaïa, Béjaïa 06000, Algérie
Tél. (213) 34 81 37 08

Résumé Le travail que nous allons présenter dans ce rapport se trouve au carrefour de trois domaines : Le clustering, la théorie des jeux et la programmation par contraintes. La particularité de ces domaines est qu'ils sont liées deux à deux. Nous nous intéressons alors à la question suivante : comment exploiter au mieux ces liens pour définir une nouvelle méthodologie efficace de clustering, basée sur des éléments de la théorie des jeux, et dont la résolution se fera moyennant les outils de la programmation par contraintes ?

Mots-clès : Jeux de potentiel, Clustering, Problème de satisfaction de contraintes, Equilibre de Nash.

4.1 Clustering et Jeux

Le problème de clustering consiste à partitionner un ensemble de données (objets) en groupes (clusters) de façon à ce que les éléments appartenant à un même cluster soient similaires mais différentes des membres des aux autres clusters.

La première application des outils de la théorie jeux au problème de clustering fait proposée par *S.R. Bulo et al* dans [1], où les auteurs ont proposé un jeu noncoopératif évolutionnaire pour la représentation du problème de clustering avec similarités asymétriques entre les éléments à regrouper. Par la suite, ils ont prouvé que les clusters formés correspondent Ã des stratégies évolutionnairement stables.

Il a été établit dans [2] que l'approche des problèmes de clustering par la théorie des jeux a pour intérÃªt principale le surpassement de quelques limitations des approches par partitionnement, tels que le chevauchement de clusters, la connaissance au préalable du nombre de clusters à former, la détection des bruits, la symétrie des matrices de similarités, etc.

Toujours dans cette dynamique d'étude des liens entre la théorie des jeux et les problèmes de clustering, nous nous somme fixé pour objectif dans ce travail, de ramener la résolution du problème de clustering de données à la résolution d'un jeu de potentiel, dont l'existence d'au moins un équilibre de Nash pure est garantie, puis démontrer que cet équilibre correspondant à un clustering stable. Pour répondre à cet objectif, nous proposons le modèle suivant :

4.1.1 Un Jeu de Potentiel pour le Clustering de Données

Considérons les éléments suivants :
- $\mathcal{O} = \{o_1, \ldots, o_n\}, o_i \in \mathbb{R}^r, i = \overline{1,n}$ un ensemble de n objets ;
- $\{C_1, \ldots, C_k\}$, l'ensemble des clusters à former à partir de $\mathcal{O}$;
- $\{c_1, \ldots, c_k\}$, leurs centroids.

En se basant sur ces éléments, nous construisons le jeu non coopératif associé, comme suit :
- Associons à chaque objet $o_i \in \mathcal{O}$, un joueur $i \in P$. $P = \{1, \ldots, n\}$ est l'ensemble des joueurs.
- Considérons que l'ensemble des strategies d'un joueur $i \in P$ coincide avec l'ensemble des clusters i.e $S_i = \{C_1, \ldots, C_k\}, \forall i \in P$.
- La fonction d'utilité que chaque joueur vise à minimiser est donnée par :

$$U_i^p(s_i, s_{-i}) = d(o_i, c_{s_i})^2 \times \alpha \tag{4.1}$$

Avec α est un paramaître qui dépend des éléments de C_{s_i}.

Le jeu correspondant au problème de clustering est alors défini par :

$$G^p = \left\langle P, \{S_i\}_{i \in P}, \{U_i^p\}_{i \in P} \right\rangle. \tag{4.2}$$

4.1.2 Quelques Résultats Théoriques

- Le jeu de clustering G^p (4.2) est un jeu de potentiel ;
- Tout jeu de potentiel fini admet au moins un équilibre de Nash en stratégies pures (PNE) ;
- Le jeu de clustering G^p admet au moins un PNE ;
- Tout équilibre de Nash pure $s^* \in S$ du jeu G^p correspond à une solution de clustering stable.

4.1.3 Résolution du Jeu de Clustering

Pour résoudre le jeu de clustering proposé, nous avons développé une nouvelle approche qui permet de rechercher un équilibre de Nash pure, en s'appuyant sur le concept de stratégies de meilleures réponses. La recherche d'une k-partition stable au sens de Nash de n objets est lancé à partir d'une solution de départ, obtenues en générant aléatoirement des centroides, puis en affectant les objets vers les centres de clusters les plus proches. Par la suite, une recherche des stratégies de meilleures réponses pour chacun des joueurs participant au jeu est faites. Si une telle stratégies existe pour un joueur et que celui-ci ne l'a pas encore choisi, ce dernier bascule vers cette stratégie, ce processus est répété jusqu'à ce qu'aucun joueur ne soit incité à modifier son choix. à ce moment là , nous auront obtenu une issue qui correspond à un clustering stable.

4.1.4 Etude expérimentale

L'algorithme proposé a été implémenté puis testé sur une série de bases de données, et les résultats obtenues ont été comparés à ceux fournis par l'algorithme classique du k-means. Ces solutions ont été évaluées sur plusieurs critères. Il a été constaté que l'algorithme proposé est le plus performant sur la plupart des tests effectués. Cependant, il présente une complexité computationnelle exponentielle. Une de nos principales perspectives est de remédier à cette insuffisance en intégrant l'aspect programmation par contraintes à notre processus de résolution.

4.2 Jeux et Contraintes

La programmation par contraintes est une discipline qui permet de modéliser des problèmes combinatoires, en le réécrivant sous forme de contraintes à satisfaire. La résolution d'un problème de PPC est prise en charge par un solveur CSP adéquat.

Un CSP $(\mathbf{P})$ défini par le triplet (X, D, C), où

- $X = \{X_i, i = \overline{1, n}\}$ est l'ensemble des variables du CSP($\mathcal{P}$).
- $D = \{D_i, i = \overline{1, n}\}$, D_i est le domaine associé à la variable X_i représentant l'ensemble de ses valeurs possibles.
- $C = \{C_1, C_2, ..., C_m\}$ est l'ensemble des m contraintes du problème $(\mathbf{P})$.

4.2.1 Exemple de Représentation d'un CSP par un Jeu Noncoopératif

La modélisation du CSP($\mathcal{P}$) par un jeu noncoopératif à n joueurs se fera comme suit :

– Associons à chaque variable X_i un joueur i. Ainsi, on aura autant de joueurs que de variables. Notons alors par $I = \{1, \ldots, n\}$ l'ensemble de ces joueurs.
– L'ensemble S_i des stratégies pures du joueur $i \in I$ est identifié à l'ensemble D_i des valeurs possibles de la variable X_i, $i \in I$. Ainsi, $S_i = D_i,\ \ i \in I$.
– Notons par $R(i)$, l'ensemble des contraintes de C liées à la variable X_i, $i \in I$. r, désigne une contrainte du CSP($\mathcal{P}$) et $k(r)$, son arité. $x = (x_1, \ldots, x_n) \in S = \prod_{i=1}^{n} S_i$, une instanciation complète des n-variables du CSP($\mathcal{P}$).

Soit la fonction indicatrice :

$$\chi_r(x_{j_1}, \ldots, x_{j_{k(r)}}) = \begin{cases} 1, \text{ si } (x_{j_1}, \ldots, x_{j_{k(r)}}) \in r, \\ 0, \text{ sinon.} \end{cases} \tag{4.3}$$

où $(x_{j_1}, \ldots, x_{j_{k(r)}}) \in r$ signifie que la contrainte r est vérifiée par l'instanciation $x = (x_1, \ldots, x_n)$ et $(x_{j_1}, \ldots, x_{j_{k(r)}})$ correspondent aux valeurs des variables intervenants dans la contrainte r.

Pour une instanciation $x = (x_1, \ldots, x_n) \in S$, on associe une utilité pour chaque joueur $i \in I$, défini par :

$$U_i(x_1, \ldots, x_n) = \sum_{r \in R(i)} k(r)\chi_r(x_{j_1}, \ldots, x_{j_{k(r)}}), \ \ \forall i \in I, \tag{4.4}$$

– On définit le jeu noncoopératif $\mathcal{G}(\mathcal{P})$ associé au problème de satisfaction de contraintes $(\mathcal{P})$ comme suit :

$$\mathcal{G}(\mathcal{P}) = < I, \{S_i\}_{i \in I}, \{U_i\}_{i \in I} >, \tag{4.5}$$

L'utilité $U_i(x_1, \ldots, x_n)$ d'un joueur i est une mesure croissante en fonction du nombre de contraintes satisfaites par la situation du jeu, sachant que les contraintes sont pondérées par leurs arités.

On peut aussi associer des poids différents $w_i(r)$ pour chaque contrainte, qui peuvent varier d'un joueur à un autre. On pourrait même introduire un facteur *perte* pour les contraintes non satisfaites. La fonction utilité s'écrira alors :

$$U_i(x_1, \ldots, x_n) = \sum_{r \in R(i)} w_i(r)\chi_r^{a,b}(x_{j_1}, \ldots, x_{j_{k(r)}}), \tag{4.6}$$

où

$$\chi_r^{a,b}(x_{j_1}, \ldots, x_{j_{k(r)}}) = \begin{cases} a \geq 0, \text{ si } (x_{j_1}, \ldots, x_{j_{k(r)}}) \in r, \\ b \leq 0, \text{ sinon.} \end{cases} \tag{4.7}$$

"a" est le gain engendré par la satisfaction d'une contrainte et "b" est la perte engendrée par sa violation.

4.3 Perspective

– Définir un modèle CSP pour la résolution des jeux de potentiels s'appuyant sur la notion de stratégies de meilleures réponses ;
– Déduire les propriétés de ce modèles (consistence,..) à partir des caractéristiques des jeux de potentiel ;
– Intégrer les contraintes de ce modèles dans un solveur CSP pour la résolution.

Références

1. Andrea Torsello, S Rota Bulā, and Marcello Pelillo. Grouping with asymmetric affinities : A game-theoretic perspective. In Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on, volume 1, pages 292-299. IEEE, 2006.
2. Pelillo Marcello. (2009). What is a cluster ? Perspectives from game theory. In Proc. of the NIPS Workshop on Clustering Theory.
3. Ricci Francesco. Equilibrium theory and constraint networks, International Conference on Game Theory, Florence, Italy, 1991.