



REVUE ALGERIENNE DES SCIENCES

Section -A-

- Sciences de la Nature et de la Vie

- Sciences techniques

<http://univ-eltarf.dz/fr/>

Hybridation PSO/ACO pour l'identification de fonction des protéines

Meriem Zekri^{a*}, Labiba Souici-Meslati^{aψ}, Karima Alem^{b,ψ}^a Laboratoire LSICO, Université Badji-Mokhtar - Annaba, B.P. 12, Annaba, 23000 Algérie.^{*} Université Chadli Bendjedid El-Tarf, B.P. 73, El-Tarf, 36000 Algérie^ψ Université Badji-Mokhtar Annaba, B.P. 12, Annaba, 23000 Algérie.^b Laboratoire BBM, Université Badji-Mokhtar-Annaba, B.P. 12, Annaba, 23000 Algérie.

Informations	Résumé
Mots clés : Hybridation PSO/ACO Biologie computationnelle Classification RCPGs Bioinformatique. *Correspondance : zekri-meriem@univ-eltarf.dz Reçu le 14 avril 2018, Révisé le 21 mai 2018, Accepté le 22 mai 2018.	Les algorithmes de la Swarm Intelligence (SI) représentent des approches bio-inspirées assez récentes qui ont fait leurs preuves dans de nombreux domaines, spécialement comme méthodes de sélection de caractéristiques. Ces dernières années, ils sont de plus en plus exploités dans le domaine de la bioinformatique qui est en lui-même un domaine récent. Les approches SI sont connues pour leur efficacité en termes de coût faible, rapidité et pour apporter des solutions raisonnablement précises. De ce fait, leur utilisation pour résoudre certaines tâches bioinformatiques s'est montrée prometteuse. Dans cet article, nous proposons d'exploiter l'algorithme hybride PSO/ACO pour la prédiction de fonction de protéine, plus précisément, des récepteurs couplés aux protéines G, qui représente l'une des tâches les plus importantes en bioinformatique.

1. Introduction

Les systèmes basés sur la swarm intelligence (SI) sont inspirés par le comportement de certains êtres vivants sociaux, tels que les fourmis, les lucioles, les oiseaux et les poissons. L'auto-organisation et le contrôle décentralisé sont des caractéristiques remarquables de ces systèmes qui, comme dans la nature, conduisent à un comportement émergent [1]. La swarm intelligence a suscité un grand intérêt durant les deux dernières décennies, et de nombreux algorithmes d'optimisation basés sur la swarm intelligence ont gagné une énorme popularité [2]. Il y a plusieurs raisons à une telle popularité, les deux principales sont probablement que ces algorithmes basés SI sont flexibles et polyvalents, et qu'ils sont très efficaces dans la résolution de problèmes non linéaires pour des applications du monde réel.

De nombreux algorithmes basés SI ont été développés et proposés pour la résolution de divers problèmes comme l'algorithme des chauves-souris, l'algorithme de pollinisation des fleurs, etc. Cependant, les algorithmes les plus populaires et les plus utilisées à ce jour restent l'optimisation par les colonies de fourmis (ACO) développé par Dorigo et al. [3] [4], et l'optimisation par les essaims particuliers (PSO) proposé par Kennedy et Eberhart [5] [6].

L'algorithme ACO s'inspire, comme son nom l'indique, du comportement des colonies de fourmis réelles pour la recherche de sources de nourriture. Plusieurs algorithmes d'optimisation par les colonies de fourmis ont été développés et présentent des performances prometteuses pour la résolution de divers problèmes dont la tâche de classification. Le PSO s'inspire quant à lui du comportement de groupes d'animaux comme les volées d'oiseaux et les bancs de poissons. Ces deux approches sont considérées comme des méthodes d'optimisation qui se basent sur la collaboration des individus entre eux, ils s'appuient également sur un système d'auto-organisation. Cette idée vient du fait qu'un groupe d'individus peu intelligents peut posséder une organisation globale complexe. En d'autres termes, ACO et PSO représentent des techniques bio-inspirées dont l'efficacité, dans la résolution de divers problèmes, a été largement prouvée. De ce fait, l'application de leur hybridation pour la tâche de classification est une voie prometteuse qui a déjà porté ses fruits dans plusieurs problèmes de classification.

Ainsi, dans cet article, nous proposons l'hybridation PSO/ACO (Hybrid PSO/ ACO) développée par Holden et Freitas [7][8], pour la tâche de classification dans le domaine de la bioinformatique, et plus précisément, celui de la prédiction de fonction des RCPGs. Les Récepteurs Couplés aux Protéines G (RCPGs) représentent

l'une des plus larges et des plus importantes familles de protéines multifonctionnelles connues de la biologie moléculaire des temps modernes. L'identification de fonctions des récepteurs couplés aux protéines G est l'un des domaines d'intérêt actuel de la recherche en bioinformatique. De ce fait, la prédiction de leur fonction se révèle être une tâche très importante à laquelle nous tentons d'apporter des solutions efficaces et peu coûteuses.

La suite de cet article est organisée comme suit. La deuxième section abordera une synthèse de travaux utilisant les approches de la swarm intelligence pour la prédiction de fonction de protéines. Dans la troisième section, nous présenterons l'algorithme hybride PSO/ACO et sa variante PSO/ACO2. Dans la quatrième section, nous définirons la famille de protéines étudiée, qui est l'une des familles les plus importantes de protéines membranaires, les récepteurs couplés aux protéines G. L'ensemble de données utilisé et les prétraitements réalisés sur ce dernier seront abordés dans la cinquième section. Dans la sixième section, nous discuterons des outils utilisés pour la réalisation des expérimentations. Les résultats et leurs comparaisons avec d'autres méthodes seront présentés dans la septième section. Enfin, nous conclurons cet article dans la huitième et dernière section avec quelques perspectives de recherche.

2. Approche de la swarm intelligence pour la prédiction de fonction de protéines

Les algorithmes de la Swarm Intelligence (SI) représentent des approches bio-inspirées assez récentes qui ont fait leurs preuves dans de nombreux domaines, spécialement comme méthodes de sélection de caractéristiques [9]. Les approches SI sont connues pour leur efficacité en termes de coût faible, rapidité et pour apporter des solutions raisonnablement précises. De ce fait, leur utilisation pour résoudre certaines tâches bioinformatiques s'est montrée prometteuse, comme pour le calcul du score de l'alignement multiple de séquences [10][11][12], la construction d'arbres phylogénétique [13] [14] et la prédiction de structure de protéines [15][16][17] [18].

Cependant, la tâche qui reste de première importance en bioinformatique est celle de la prédiction de fonctions de protéines. Diverses méthodes SI ont été exploitées afin d'y apporter des solutions optimales. Parmi ces méthodes, l'algorithme hybride PSO/ACO (version originale) proposé par Holden et Freitas [8] pour une classification hiérarchique des fonctions de protéines en utilisant les bases de données UniProt et GPCRDDB afin de construire leur ensemble de données. Ils ont proposé cette hybridation pour pallier aux inconvénients des deux méthodes ACO et PSO utilisées séparément.

Ils ont comparé l'algorithme hybride PSO/ACO avec l'algorithme DPSO (Discrete Particle Swarm Optimization) et les résultats fournis montrent que leur méthode a apporté des solutions plutôt prometteuses.

Correa et al. [19] ont utilisé l'algorithme DPSO dans l'étape de sélection des caractéristiques pour une classification de fonctions de protéines par un réseau Bayésien. L'étape de sélection de caractéristiques joue un rôle très important dans la prédiction de fonctions de protéines, elle permet de représenter les séquences protéiques sans perdre les informations essentielles. Elle aide donc à améliorer la précision prédictive et une méthode de sélection de caractéristique inappropriée peut affecter les résultats finaux. Dans leur cas, l'algorithme DPSO a amélioré considérablement la précision prédictive de leurs classifieurs en comparaison à l'utilisation de tous les attributs.

Une autre approche SI pour la sélection de caractéristiques a été développée par Nemati et al. [20]. Elle est basée sur l'algorithme hybride ACO-GA (Ant Colony Optimization – Genetic Algorithm). Cet algorithme hybride a été développé dans le cadre de la prédiction de fonctions de protéines afin de faire face au grand nombre d'attributs des séquences protéiques. Afin de tester leur algorithme hybride, Nemati et al. ont utilisé deux ensembles de données, l'un comportant des séquences de récepteurs couplés aux protéines G (RCPGs) et l'autre comportant des séquences d'enzymes. Les résultats obtenus montrent l'amélioration de la prédiction prédictive après l'application de cette méthode de sélection de caractéristiques.

Gu et Ding [21] ont présenté l'algorithme des essaims particuliers binaires (Binary Particle Swarm Intelligence – BPSO) pour extraire les caractéristiques pertinentes des compositions de paires d'acides aminés des séquences protéiques RCPGs. Ils ont utilisé ensuite l'approche des k-plus proches voisins pour réaliser la classification. Ils ont comparé leur méthode de sélection de caractéristiques avec la composition en pseudo acides aminés (PseAAC) proposée par Chou [22], et avec la composition en acides aminés (AAC) proposée par Guo et al. [25]. Bien que leur méthode soit encourageante, c'est la méthode PseAAC qui a fourni les meilleurs résultats, du moins au niveau de la famille A.

Hoden et Freitas [23] ont proposé une nouvelle méthode appelée HEHRS (Hierarchical Ensemble of Hierarchical Rules Set) pour le problème de la classification hiérarchique. Cette méthode combine la prédiction d'un ensemble de classifieurs afin d'accroître la précision prédictive. Hoden et Freitas ont utilisé l'approche PSO pour optimiser la création des règles et ont testé leur méthode sur six ensembles protéiques en utilisant les signatures protéiques obtenues

après une collecte de l'ensemble de données de GCPDB qu'ils ont ensuite scanné afin d'obtenir les régions conservées (motifs et signatures). Comme nous pouvons le constater à travers cette synthèse de travaux, il y a un manque de travaux utilisant les approches de la swarm intelligence pour l'identification de fonctions de protéines et la majorité des travaux, publiés dans la littérature du domaine, utilisent plutôt la swarm intelligence pour l'étape de sélection des caractéristiques que pour celle de la classification.

3. L'hybridation PSO/ACO

L'algorithme hybride PSO/ACO a été développé par Holden et Frietas [7] pour la découverte de règles de classification dans le contexte de la fouille de données, plus précisément, pour la découverte de nouvelles protéines et leurs fonctions. Cependant, l'algorithme PSO/ACO est un algorithme générique qui peut être appliqué à d'autres problèmes de classification.

Bien que les algorithmes ACO et PSO aient été appliqués avec succès à divers problèmes de classification, Holden et Freitas ont pensé qu'ils ne sont pas si parfaitement adaptés à ces problèmes. La première motivation pour la conception de cette hybridation était de rendre l'algorithme PSO plus efficace dans le traitement des attributs nominaux en utilisant le mécanisme basé sur les phéromones de l'algorithme ACO. L'algorithme ACO a également certains inconvénients comme un temps de calcul assez élevé par rapport aux méthodes de classification connues et ceci en raison de la procédure d'élagage de la règle qui est nécessaire à chaque itération interne de l'algorithme. Aussi, le fait que l'algorithme traite des attributs continus qui doivent être discrétisés au préalable, peut diminuer la précision de la classification et augmenter le temps de calcul avec cette étape de prétraitement.

3.1 Formalisme

L'algorithme PSO/ACO utilise une approche séquentielle de couverture pour découvrir une règle de classification à la fois. La première étape consiste en l'initialisation de l'ensemble de règle (Rule Set – RS) avec un ensemble vide. Après, pour chaque niveau hiérarchique de classe et pour chacune des classes à prédire, l'algorithme effectue une boucle « Tant que ».

```
RS = {} \* initialement, l'ensemble de règles est vide
Pour chaque niveau L
  Pour chaque class C de niveau L
    TS = {tous les exemples de l'apprentissage appartenant aux classes au niveau L avec le même parent C};
    Tant que (nombre d'exemples non-couverts de C > MaxUncovExampClass)
```

```
Exécuter l'algorithme PSO/ACO pour
découvrir la meilleure règle qui prédit la classe
C, appelée meilleure classe ;
RS = RS ∪ meilleure règle ;
TS = TS - {exemples d'apprentissage
correctement couverts par la règle
découverte}
  Fin Tant que
    Fin pour
  Fin pour
```

Algorithme 1 : Approche séquentielle de couverture par l'algorithme hybride PSO/ACO [7].

Chaque itération de cette boucle effectue une exécution de l'algorithme PSO/ACO, retournant la meilleure règle découverte prédisant les exemples de la classe courante C. Cette règle est ajoutée à l'ensemble de règle (RS) et les exemples correctement couverts par la règle sont retirés de l'ensemble d'apprentissage (Training Set – TS). Un exemple est dit correctement couvert par une règle si cet exemple satisfait tous les termes (les paires « attributs – valeurs ») dans l'antécédent de la règle et il appartient à la classe prédite par la règle. Cette boucle « Tant que » est effectuée aussi longtemps que le nombre d'exemples non-couverts de la classe C est supérieur à un seuil défini par l'utilisateur, le nombre maximal d'exemples non-couverts par classe (MaxUncovExampClass). Après la découverte des règles pour toutes les classes à tous les niveaux, l'algorithme retourne RS, l'ensemble de règles découvertes.

Chaque particule représente l'antécédent de la règle de classification candidate. La classe de la règle est fixée par toutes les particules lors de chaque exécution de l'algorithme et chaque exécution de l'algorithme vise à découvrir la meilleure règle pour une classe précise.

3.2 Version améliorée de l'hybridation PSO/ACO : PSO/ACO2

Dans notre proposition, nous avons utilisé la version modifiée de l'algorithme hybride conventionnel PSO/ACO développée également par Holden et Frietas [24]. Pour l'essentiel, les modifications apportées impliquent des changements dans la procédure de mise à jour des phéromones et dans l'étape d'initialisation de l'ensemble de règles, ainsi que de manière significative, le fractionnement du processus de découverte des règles en deux phases distinctes, comme suit : dans la première phase, une règle est découverte en utilisant les attributs nominaux seulement. Dans la seconde phase, la règle est potentiellement étendue avec les attributs continus. Cela augmente la capacité de l'algorithme hybride PSO/ACO2 dans le traitement des attributs nominaux et continus de différentes manières, tout en reconnaissant la

différence entre ces deux types d'attributs (un fait ignoré par l'algorithme PSO/ACO conventionnel).

Les deux versions de l'algorithme hybride PSO/ACO utilisent l'approche séquentielle de couverture pour découvrir une règle de classification à la fois. Comme pour la première version, PSO/ACO2 commence par l'utilisation d'un ensemble de règles vide. Ensuite, pour chaque classe, l'algorithme effectue la boucle « Tant que », l'ensemble TS est utilisé pour stocker les exemples d'apprentissages à partir desquels les règles seront construites.

```

RS = {} \* initialement, l'ensemble de règles
est vide
Pour chaque classe C
  TS = {tous les exemples d'apprentissage}
  Tant que (nb exemple d'apprentissage non-
  couverts de la classe C >
  MaxUncovExampPerClass)
    Exécuter PSO/ACO pour découvrir la
    meilleure règle nominale prédisant C
    Exécuter PSO standard pour ajouter les
    termes continus à la règle et retourner la
    meilleure règle, BestRule
    Elaguer BestRule
    RS = RS ∪ BestRule
    TS = TS - {exemples correctement
    couverts par la règle découverte}
  Fin TQ
Fin pour
Mettre les règles en RS par ordre décroissant
de qualité

```

Algorithme 2 : Approche séquentielle de couverture par l'algorithme hybride PSO/ACO2 [24].

Chaque itération de cette boucle réalise une exécution de l'algorithme PSO/ACO2 qui découvre uniquement des règles basées sur les attributs nominaux, retournant la meilleure règle. La règle retournée par PSO/ACO2 est souvent incomplète parce qu'elle ne comprend pas de termes de valeurs continues. Cette règle est ensuite utilisée comme base pour la découverte de termes avec des valeurs continues. L'algorithme standard PSO (appliqué uniquement aux attributs numériques) est utilisé avec constrictif. Le vecteur à optimiser est constitué de deux dimensions par attribut continu, la première pour la limite supérieure et la seconde pour la limite inférieure.

À chaque évaluation de particules le vecteur est transformé en un ensemble de termes (conditions de la règle) et est ajouté à la règle produite par l'algorithme PSO/ACO2 pour l'évaluation par fitness. Par exemple, si l'ensemble de données contenait un attribut nominal A_{n0} et un attribut continu A_{c0} , l'algorithme PSO/ACO2 pourrait produire une

règle comme : Si A_{n0} = "valeur" alors classe C. L'algorithme standard PSO serait alors tenté d'améliorer cette règle en ajoutant des termes : $x_{i1} > A_{c0}$ et $x_{i2} \leq A_{c0}$, ce qui correspond effectivement à un terme de la forme : $x_{i1} > A_{c0} \geq x_{i2}$. Où la position d'une particule serait le vecteur x_{i1}, x_{i2} . La règle pour l'évaluation serait :

Si $A_{n0} = \text{< valeur >}$ et $x_{i1} > A_{c0}$ et $x_{i2} \leq A_{c0}$
alors classe C

Si les deux bornes croisent les deux termes $0 < A_{c0} < 1$, les deux sont retirés de la règle mais la meilleure position (P_i) est toujours mise à jour dans ces dimensions :

$$v_{id} = \chi \left(x_{id} + c_1 \epsilon_1 (p_{id} - x_{id}) + c_2 \epsilon_2 (p_{gd} - x_{id}) \right) \quad (1)$$

$$x_{id} = x_{id} + v_{id} \quad (2)$$

La formule 1 concerne la mise à jour de la vitesse de PSO et la formule 2 concerne la mise à jour de la position de PSO. La meilleure règle est alors ajoutée à l'ensemble de règles après avoir été élaguée en utilisant la méthode d'élagage d'Ant-Miner, et les exemples correctement couverts par cette règle sont retirés de l'ensemble d'apprentissage (TS). Un exemple est dit correctement couvert par une règle si cet exemple satisfait tous les termes (les paires « attribut-valeur ») dans l'antécédent de la règle et il a la classe prédite par la règle. La boucle « Tant que » est exécutée aussi longtemps que le nombre d'exemples non couverts de la classe C est supérieur à un seuil défini par l'utilisateur, qui représente le nombre maximal d'exemples non couverts par classe (MaxUncovExampPerClass). Avoir un ensemble de règles non ordonnées est important parce que, après la création de l'ensemble de règles, les règles sont ordonnées par qualité et non par ordre de création. Bien que ce soit une approche gloutonne, elle a amélioré la précision prédictive dans des expériences antérieures.

4. Les récepteurs couplés aux protéines G

Les récepteurs couplés aux protéines G servent de médiateurs essentiels pour divers événements de transduction du signal cellulaire qui permet aux cellules, tissus, organes et à l'organisme, en entier, de réagir correctement aux changements environnementaux exigés. Les fonctions des RCPGs sont extrêmement diverses vu qu'ils régulent de nombreux processus physiologiques liés aux fonctions neurologiques et neurodégénératives, cardiovasculaires et aux mécanismes de contrôle métabolique. Le nombre estimé des RCPGs dans le génome humain est de 800, ce qui représente approximativement 3% du génome humain [26][27].

3.1 Structure des RCPGs

Les récepteurs couplés aux protéines G sont des protéines membranaires caractérisées par sept hélices transmembranaires (TM), ils répondent à différents ligands, tels que les acides aminés, les peptides, les hormones et les substances odorantes, à partir du côté extracellulaire de la cellule. Ces stimulations induisent les RCPGs à activer les protéines G et à transmettre les signaux à l'intérieur de la cellule, les RCPGs existent dans la plupart des cellules et la transduction d'un signal anormal de RCPGs est liée à diverses conditions anormales telles que l'allergie, les troubles cardiaques, le cancer, l'hypertension artérielle, etc. [29]. La taille exacte de la superfamille des RCPG est inconnue, mais près de 800 différents gènes humains ont été calculées à partir de l'analyse de la séquence du génome. La masse moléculaire des RCPGs varie pour la plupart de 40 à 85 kda (kilo dalton). L'extrémité amine-terminale de la protéine est extracellulaire. L'extrémité carboxy-terminale est intracellulaire. On peut observer trois boucles extracellulaires (nommées E1, E2 et E3) et trois boucles intracellulaires (I1, I2 et I3) (voir Figure 1).

Ces protéines peuvent être sujettes à des modifications post-traductionnelles, de type N-glycosylation, acylation par des composés lipidiques (formant parfois une pseudo-

quatrième boucle intracellulaire (I4), formation de ponts disulfures entre les chaînes latérales de deux résidus de Cystéine. Le RCPG possède une analogie faible à bacterio-rhodopsin pour laquelle une structure a été déterminée par diffraction d'électrons (PDB, 2BRD, 1AT9) et X-rayon basée cristallographie (1AP9). En 2000, la première structure cristalline d'un RCPG de mammifères a été résolue, la rhodopsine bovine (1F88). La structure des extrémités N- et C-terminal des RCPGs peuvent également remplir d'importantes fonctions au-delà de la liaison de ligand, en particulier, l'extrémité C-terminal qui contient le plus souvent la sérine et la thréonine ; des résidus qui, lorsqu'ils sont phosphorylés, augmentent l'affinité de la surface intercellulaire pour la liaison des protéines d'échafaudage appelées β -arrestines. Une fois liées, les β -arrestines préviennent à la fois le couplage des protéines G et recrutent d'autres protéines conduisant à la création d'une signalisation complexe impliquée dans l'activation du signal extracellulaire régulé kinase ou le récepteur de l'endocytose (internalisation). Comme la phosphorylation des résidus des sérines et des thréonines se produit souvent à la suite de l'activation des récepteurs couplés aux protéines G, les β -arrestines méditées aux couplages des protéines G et l'internalisation des RCPGs sont des mécanismes importants de la désensibilisation [29].

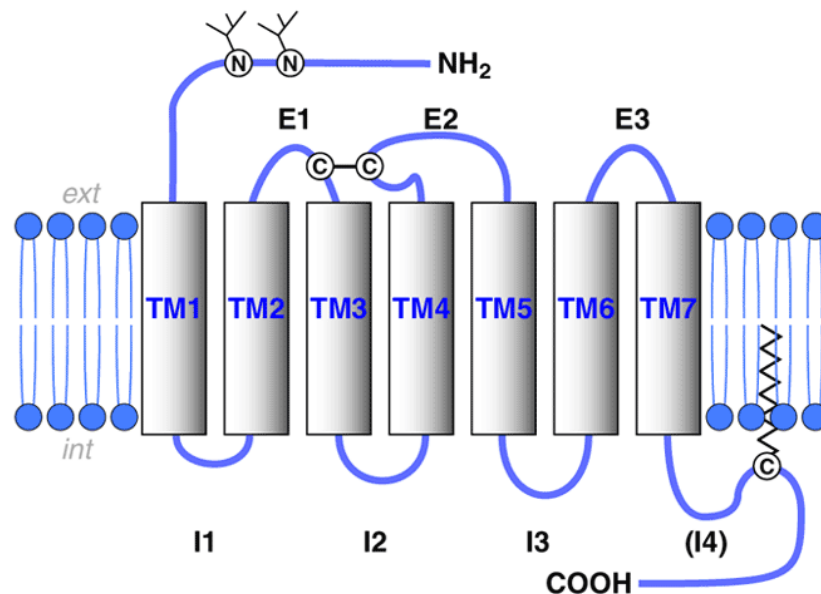


Fig. 1 : Structure de la majorité des récepteurs couplés aux protéines G [30].

3.1 Classification des RCPGs

Sur la base de connaissances pharmacologiques, le système d'information GPCRDB [31][32][33] organise la superfamille des RCPG en une hiérarchie de familles, sous-familles, sous-sous-familles et types (voir Figure 2). En tout, les RCPGs peuvent être regroupées en 6 classes

basées sur une homologie de séquence et une similitude fonctionnelle. Malgré le manque d'homologie de séquence entre les classes, tous partagent une structure commune et le mécanisme de transduction du signal. La classification généralement acceptée pour les RCPGs chez les vertébrés est la suivante.

Rhodopsine (famille A), Sécérétine (Famille B), Glutamate (famille C), Adhérence (famille D), et la famille E et Frizzled/Taste. Cependant, la particularité des éléments structurels qui les

caractérisent ne distinguent que les trois premières familles (A, B et C) comme étant principales.

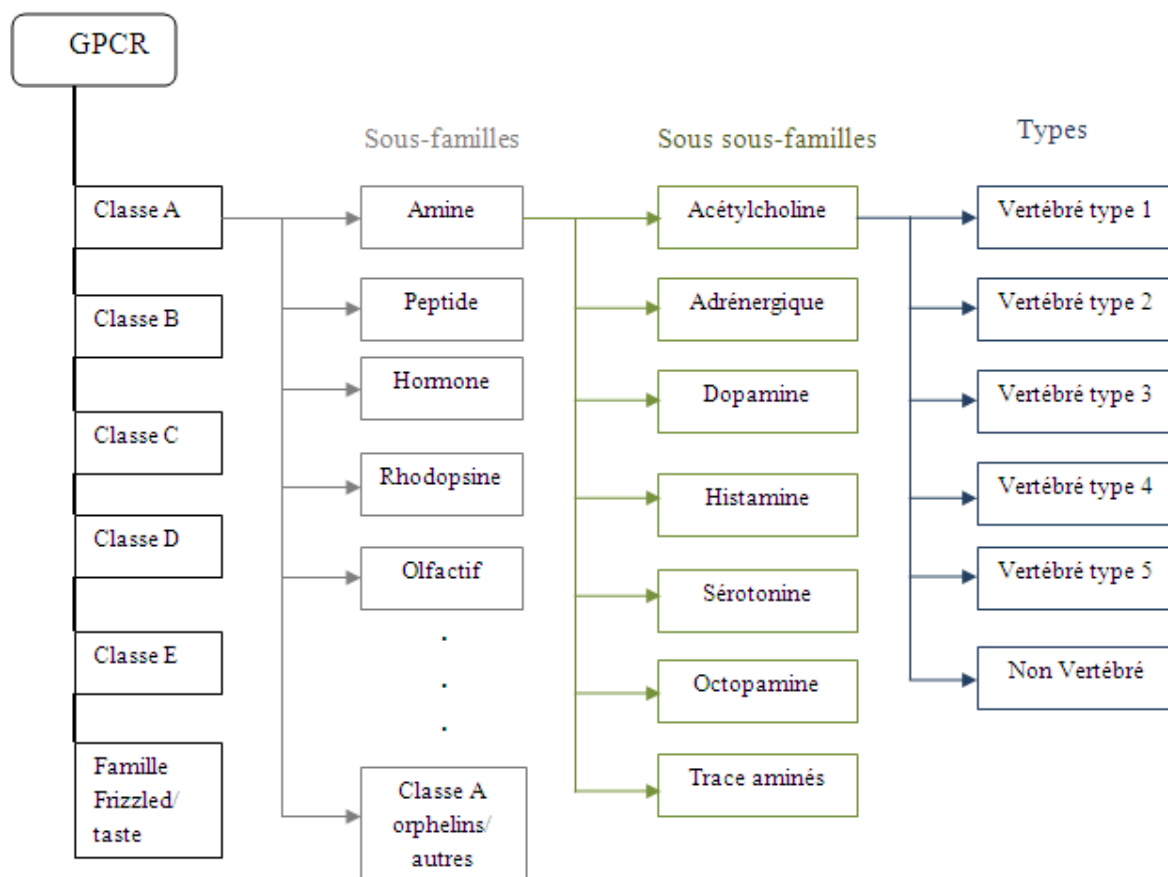


Fig. 2 : Vue simplifiée de l'arbre des familles des récepteurs couplés aux protéines G (classification du système d'information GPCRDB) [34].

4. Ensemble de données et prétraitements

Il y a diverses bases de données RCPGs [35][36] et différents ensembles de données ont été construits à partir de ces bases, tels que GDS [37], GDFL [38], D167 [39] et D365 [40]. Parmi ces ensembles de données, nous avons choisi l'ensemble GDS parce qu'il représente l'un des ensembles de données les plus largement utilisés dans le domaine de l'identification des RCPGs. L'ensemble de données GDS comprend 8.222 séquences protéiques RCPGs identifiées en utilisant l'outil Entrez Search et Retrieval System qui recherche les bases de données comme SwissProt, PIR (Protein Information Resource), PRF (Protein Research Foundation), PDB (Protein Data Base), aussi bien que les bases de données de transcription des régions d'ADN codées et annotées telles que GenBank et RefSeq (Reference Sequence Database) [41]. Toutes les séquences sont de plus de 280 résidus (acides aminés) de long et seules les classes contenant 10 séquences et plus sont retenues. Dans notre étude, nous réalisons l'identification des quatre premiers niveaux seulement en raison de

l'absence du type (niveau cinq) dans l'ensemble de données GDS.

Nous avons utilisé deux étapes de prétraitement, qui consistent en une sélection de caractéristiques utilisant la méthode de la composition en pseudo acides aminés (PseAAC) de type 1 qui génère des vecteurs de caractéristiques de plus petites dimensions [42][43] au lieu du type 2. Ce choix se justifie par notre souci d'améliorer la précision prédictive et le fait que l'approche proposée ne peut gérer correctement des espaces de recherches de grandes dimensions, il est donc important de réduire la dimensionnalité tout en gardant le plus d'informations pertinentes. La seconde étape consiste aussi en une normalisation mais la méthode choisie est celle de la normalisation Min/ Max qui représente l'une des méthodes les plus populaires. La différence entre la PseAAC type 1 et type 2 est que le type 2 génère des vecteurs de dimension $(20+i \times \lambda)$ tandis que le type 1 génère des vecteurs de dimension $(20+\lambda)$.

6. Outils pour les expérimentations

Dans cette étude qui consiste en l'identification de fonctions de récepteurs couplés aux protéines

G (RCPGs), nous proposons un classifieur basé sur l'hybridation PSO/ACO (PSO/ACO2). Ce classifieur va être testé en utilisant l'ensemble de données GDS afin de pouvoir comparer les performances de ce classifieur avec le classifieur immunologique AIRS proposé dans des travaux antérieurs et également avec des méthodes publiées. Dans cette section, nous allons aborder les outils utilisés pour l'implémentation du classifieur proposé.

6.1 PSO/ACO2

L'outil PSO/ACO2 est utilisé pour pouvoir tester l'algorithme hybride avec notre ensemble de données. PSO/ACO2 [24] est un logiciel libre et gratuit développé en langage Java, il fournit un niveau spécialisé de la fouille de données pour appliquer la technique PSO/ACO au problème de classification avec un choix à faire entre PSO standard ou DE pour l'optimiseur continu (pour la construction de la règle à partir des attributs continus). Les paramètres de l'algorithme sont dans le tableau qui suit.

Paramètre	Valeur défaut	par
validation croisée	10	
Nombre de particules	10	
Nombre d'itération	200	
Fonction de fitness	Précision	
Optimiseur continu	PSO (ou DE)	

Tableau 1 : Paramètres de l'algorithme PSO/ACO2.

6.2 PseAAC server

Pour la sélection des caractéristiques, nous avons le choix entre trois outils de la composition en pseudo acide aminée (PseAAC), PseAAC server [45] qui est un serveur en ligne qui nous permet de soumettre des séquences protéiques et nous renvoie des vecteurs de différentes dimensions selon les paramètres choisies, l'inconvénient cependant avec cet outil, est que ne nous pouvons soumettre que 500 séquences à la fois, ce qui s'avère très contraignant lorsque l'on doit soumettre des milliers de séquences. Il y a également PseAAC Builder [46], plus récent et plus avantageux dans le sens où il n'y a pas de limite de nombre de séquences à soumettre et aussi c'est un outil à installer sur l'ordinateur, de ce fait, il n'est pas nécessaire d'avoir une connexion Internet pour l'utiliser, en plus du fait qu'il soit plus rapide. Enfin, il y a Propy [47], un autre outil pour générer la composition en pseudo acides aminés qui est semblable à PseAAC-BUILDER. Nous avons utilisé PseAAC server bien qu'il présente un inconvénient. Ceci

parce qu'au moment où nous avons effectué nos expérimentations, nous n'avions pas connaissances des autres outils. Nous avons donc sélectionné PseAAC type 1 et toutes les caractéristiques des acides aminés, c'est-à-dire, l'hydrophobie, l'hydrophilie, la fonction carboxylique (-COOH), la fonction amine (-NH₃), la masse molaire et le point isoélectrique (PI) à 25°C (par défaut). Nous avons fixé le paramètre lambda à 1 et le facteur poids à 0.05 (valeurs par défaut). Avec ce jeu de paramètres, nous avons obtenu des vecteurs de 22 dimensions (moins volumineux que si l'on avait choisi le type 2). Suite à cela, nous avons réarrangé l'agencement de ces vecteurs afin de pouvoir les exploiter par le classifieur proposé.

7. Résultats

Le Tableau qui suit montre les résultats du classifieur proposé, basé sur l'hybridation PSO/ACO2. Les résultats ne sont pas aussi prometteurs que nous le souhaitons. Ceci s'explique par le fait que lorsque l'espace de recherche est très grand, la possibilité de trouver des solutions optimales se complique, bien que nous ayons réduit l'espace de recherche à des vecteurs de 21 dimensions et réduit également l'intervalle de ces vecteurs à [0,1], les résultats restent moins bons que ceux que nous avons obtenus avec le classifieur immunitaire AIRS. Cependant, nous remarquons qu'aux niveaux inférieurs les résultats sont plus probants qu'aux niveaux supérieurs bien que cela s'explique par le fait qu'à ces niveaux, les ensembles de données sont moins volumineux, ce qui facilite la classification.

	PSO/ACO2
Superfamille	97.03%
Familles	92.80%
Sous-familles	81.56%
Sous-sous-familles	93.75%

Tableau 2 : Précision prédictive du classifieur PSO/ACO2.

7.1 Comparaison avec les classifieurs immunitaires

Nous avons comparé les résultats obtenus par notre classifieur basé swarm (PSO/ACO2) avec ceux du classifieur immunitaire AIRS2 [48] en utilisant le même ensemble de données GDS et les mêmes prétraitements de la méthode PseAAC type 1 et la normalisation Min/Max, utilisées par le classifieur basé swarm, afin de déterminer si le type de prétraitement affecte les résultats d'AIRS2.

PSO/ACO2	AIRS2	AIRS2
----------	-------	-------

	(avec PseAAC type 1)	(avec PseAAC type 1)	(avec PseAAC type 2)
<i>Superfamille</i>	97.03%	98.20%	98.00%
<i>Familles</i>	92.80%	97.80%	97.50%
<i>Sous-familles</i>	81.56%	89.98%	89.20%
<i>Sous-sous-familles</i>	93.75%	91.03%	90.95%

Tableau 3 : Comparaison de la précision prédictive des meilleurs classifieurs bio-inspirés PSO/ACO2 et AIRS2.

Lors de nos premières expérimentations, nous avons choisi la méthode PseAAC type 2 pour la représentation des séquences de RCPGs qui rappelons-le, produit des vecteurs de caractéristiques de dimension $(20+i \times \lambda)$ où le nombre 20 représente les 20 acides aminés natifs, i représente le nombre des attributs d'acides aminés sélectionnés et λ est le paramètre qui permet de ne pas perdre l'ordre des acides aminés dans la séquence. Ce type de la méthode de la PseAAC est celui qui nous permet de construire des vecteurs de caractéristiques de dimension convenable, tout en gardant le plus d'informations pertinentes. Au contraire, la méthode PseAAC type 1 bien qu'elle soit efficace également, elle ne fournit pas les mêmes performances que la PseAAC type 2 car elle ne permet pas de prendre en compte le nombre des attributs d'acides aminés sélectionnés, quel que soit ce que nous sélectionnons comme caractéristique, la dimension du vecteur reste inchangée. Cependant, elle représente un avantage, celui de produire des vecteurs de petite dimensionnalité (de dimension $(20+\lambda)$) que l'on sélectionne toutes les caractéristiques d'acides aminés (l'hydrophobie, l'hydrophilie, la fonction carboxylique (-COOH), la fonction amine (-NH₃), la masse molaire et le point isoélectrique (PI) à 25°C (par défaut)) ou certaines seulement. Pour les expérimentations avec l'approche de la swarm intelligence, nous étions obligés de choisir la PseAAC type 1, car la méthode sélectionnée ne

pouvait pas traiter efficacement des ensembles de grande dimensionnalité.

Nous avons testé à nouveau le classifieur AIRS2 avec le nouvel ensemble de données obtenu avec la méthode PseAAC type 1 (voir Tableau 3). Nous remarquons que le fait de réduire la dimensionnalité de l'ensemble améliore les résultats (par rapport à l'utilisation de méthode PseAAC type 2) même si cette amélioration n'est pas vraiment très significative.

7.2 Comparaison avec les méthodes publiées

Dans la littérature, peu de classifieurs basés sur les approches de la swarm intelligence (SI) ont été développés dans le cadre de la prédiction de fonction des récepteurs couplés aux protéines G. Pour une validation supplémentaire des résultats de notre classifieur basé SI, nous avons réalisé une comparaison avec des méthodes publiées : l'hybridation PSO/ACO de Holden et Freitas [8], l'algorithme PSO binaire de Gu et Ding [21] et l'approche hybride PSO-HERHS de Holden et Freitas [23], qui abordent la même problématique d'identification des RCPGs mais en utilisant des ensembles de données différents, collectés à partir des bases de données UniProt et GPCRDB. Bien que cette comparaison ne soit pas tout à fait pertinente vu que les ensembles de données ne sont pas les mêmes, elle est réalisée afin de comparer les performances des approches SI de manière générale à traiter la problématique posée.

	<i>superfamille</i>	<i>Familles</i>	<i>Sous-familles</i>	<i>Sous-sous-familles</i>
PSO/ACO Holden et Freitas [8]	89.46%	63.44%	45.18%	33.76%
PSO binaire Gu et Ding [21]	–	98.02%	–	–
PSO-HERHS Holden et Freitas [23]	91.5%	82.09%	57.43%	86.03%
PSO/ACO2	97.03%	92.80%	81.56%	93.75%

Tableau 4. Comparaison de la précision prédictive de PSO/ACO2 avec des méthodes publiées.

Dans le tableau 4, nous constatons qu'au niveau famille c'est le classifieur de Gu et Ding [21] qui a fourni les meilleurs résultats avec une précision de 98.02%. Leur classifieur semble avoir fourni de meilleures performances, connues à ce jour, au

niveau famille, en comparaison aux méthodes publiées selon notre connaissance. Nous remarquons également que les travaux de Holden et Freitas [8][23] ont fourni des résultats moins probants que les nôtres et ce à tous les niveaux,

ce qui confirme que bien que les approches de la swarm intelligence aient fait leurs preuves dans bien des domaines de classification, elles semblent être moins appropriées pour l'identification des RCPGs, en comparaison à d'autres approches.

8. Conclusion

La Swarm Intelligence représente une approche bio-inspirée assez récente et prometteuse dans de nombreux domaines d'application. Dans cet article, nous avons proposé un classifieur basé sur l'hybridation PSO/ACO2, pour le problème de prédiction de fonctions de récepteurs couplés aux protéines G (RCPGs), en utilisant l'ensemble de données GDS et la méthode de composition en pseudo acides aminés (PseAAC) de type 1 pour la sélection des caractéristiques.

Les expérimentations effectuées ne nous ont pas fourni les résultats escomptés, et ceci pour deux raisons, la première est la grande dimensionnalité de l'espace de recherche, bien que nous ayons tenté de la réduire en utilisant la PseAAC. La seconde est le nombre important de classes à certains niveaux qui a rendu la tâche de classification plus difficile à réaliser. L'hybridation PSO/ACO2 reste un modèle qui a apporté des résultats intéressants bien qu'ils restent insuffisants en comparaison au classifieur immunitaire (AIRS2) présenté dans nos travaux précédents. L'hybridation a fourni de meilleures performances car elle a été conçue pour pallier aux inconvénients des deux modèles ACO et PSO utilisés séparément. Un autre inconvénient s'ajoute aux résultats, celui d'un temps d'exécution prohibitif, particulièrement, aux niveaux supérieurs en raison du gros volume des ensembles d'apprentissage.

L'approche de la Swarm Intelligence reste tout de même une approche intéressante pour le domaine de la fouille de données en bioinformatique si l'on trouve le moyen d'améliorer les performances en utilisant, ou en hybridant, d'autres méthodes de cette catégorie, ou bien le moyen de réduire encore plus les espaces de recherche tout en conservant les informations pertinentes.

Référence :

- [1] Parpinelli, R.S., Lopes, H.S.: New Inspirations in Swarm Intelligence: A Survey, *International Journal of Bio-Inspired Computation*, vol. 3, issue 1, pp. 1–16, 2011.
- [2] Yang, X.S., Karamanoglu, M.: *Swarm Intelligence and Bio-Inspired Computation: An Overview*, in: Yang, X.S., Cui, Z., Xiao, R., Gandomi, A.H., Karamanoglu, M. (Eds.), *Swarm intelligence and bio-inspired computation: theory and applications*, Elsevier, pp. 3–23, 2013.
- [3] Dorigo, M., Gambardella, L.M.: Ant Colonies for the Travelling Salesman Problem, *BioSystems*, vol. 43, issue 2, pp. 73–81, 1997.
- [4] Dorigo, M., Stützle, T.: *The Ant Colony Optimization Metaheuristic: Algorithms, Applications, and Advances*, in F. Glover, G.A. Kochenberger (Eds.), *Handbook of Metaheuristics*, International Series in Operations Research & Management Science, 57, Springer US, pp. 250–285, 2003.
- [5] Kennedy, J., Eberhart, R.: Particle Swarm Optimization, *IEEE International Conference on Neural Networks*, vol. 4, pp. 1942–1948, 1995.
- [6] Poli, R., Kennedy, J., Blackwell, T.: Particle Swarm Optimization, *Swarm Intelligence*, vol. 1, issue 1, pp. 33–57, 2007.
- [7] Holden, N., Freitas, A.A.: A Hybrid Particle Swarm/Ant Colony Algorithm for the Classification of Hierarchical Biological Data, *IEEE swarm intelligence symposium (SIS)*, pp. 100–107, 2005.
- [8] Holden, N., Freitas, A.A.: Hierarchical Classification of G-Protein-Coupled Receptors with a PSO/ACO Algorithm, *IEEE Swarm Intelligence Symposium (SIS)*, pp. 77–84, 2006.
- [9] Das, S., Abraham, A., Konar, A.: *Swarm Intelligence Algorithms in Bioinformatics*, in A. Kelemen, A. Abraham, Y. Chen (Eds.), *Computational Intelligence in Bioinformatics, Studies in Computational Intelligence*, 94, Springer Berlin Heidelberg, pp. 113–147, 2008.
- [10] Rasmussen, T.K., Krink, T.: Improved Hidden Markov Model Training for Multiple Sequence Alignment by a Particle Swarm Optimization—Evolutionary Algorithm Hybrid. *Biosystems*, vol. 72, issue 1, pp. 5–17, 2003.
- [11] Chan, A., Freitas, A.A.: A New Ant Colony Algorithm for Multi-label Classification with Applications in Bioinformatics, *Annual Conference on Genetic and Evolutionary Computation (GECCO)*, ACM, pp. 27–34, 2006.
- [12] Lalwani, S., Kumar, R., Gupta, N.: A Novel Two-level Particle Swarm Optimization Approach for Efficient Multiple Sequence Alignment, *Memetic Computing*, pp. 1–15, 2015.
- [13] Perretto, M., Lopes, H.S.: Reconstruction of Phylogenetic Trees using the Ant Colony Optimization Paradigm, *Genetics and Molecular Research*, vol. 4, issue 3, pp. 581–589, 2005.
- [14] Dinh, H.Q., Minh, B.Q., Huan, H.X., Von Haeseler, A.: ACOPHY: A Simple and General Ant Colony Optimization Approach for Phylogenetic Tree Reconstruction, in M. Dorigo et al. (Eds.), *Swarm Intelligence, Lecture Notes in Computer Science*, 6234, Springer Berlin Heidelberg, pp. 360–367, 2010.
- [15] Chu, D., Zomaya, A.: Parallel Ant Colony Optimization for 3D Protein Structure Prediction using the HP Lattice Model, in N. Nedjah, L.M. Mourelle, E. Alba (Eds.), *Parallel Evolutionary Computations, Studies in Computational Intelligence*, 22, Springer Berlin Heidelberg, pp. 177–198, 2006.
- [16] Fidanova, S.: 3D HP Protein Folding Problem using Ant Algorithm, *BioPS International Conference*, pp. III.3–III.26, 2006.
- [17] Mansour, N., Kanj, F., Khachfe, H.: Particle Swarm Optimization Approach for Protein Structure Prediction in the 3D HP Model, *Interdisciplinary*

- Sciences: Computational Life Sciences, vol. 4, issue 3, pp. 190–200, 2012.
- [18] Alqattan, Z.N., Abdullah, R.: A Comparison Between Artificial Bee Colony and Particle Swarm Optimization Algorithms for Protein Structure Prediction Problem, in M. Lee, A. Hirose, Z.G. Hou, R.M. Kil (Eds.), *Neural Information Processing, Lecture Notes in Computer Science*, 8227, Springer Berlin Heidelberg, pp. 331–340, 2013.
- [19] Correa, E.S., Freitas, A.A., Johnson, C.G.: Particle Swarm for Attribute Selection in Bayesian Classification: An Application to Protein Function Prediction, *Journal of Artificial Evolution and Applications*, vol. 17, 2008.
- [20] Nemati, S., Basiri, M.E., Ghasem-Aghaee, N., Aghdam, M.H.: A Novel ACO–GA Hybrid Algorithm for Feature Selection in Protein Function Prediction, *Expert Systems with Applications*, vol. 36, issue 10, pp. 12086–12094, 2009.
- [21] Gu, Q., Ding, Y.: Binary Particle Swarm Optimization based Prediction of G-Protein-coupled Receptor Families with Feature Selection, *ACM/SIGEVO Summit on Genetic and Evolutionary Computation (GEC)*, pp. 171–176, 2009.
- [22] Chou, K.C.: Prediction of G-protein-coupled receptor classes, *Journal of Proteome Research*, vol. 4, issue 4, pp. 1413–1418, 2005.
- [23] Holden, N., Freitas, A.A.: Hierarchical Classification of Protein Function with Ensembles of Rules and Particle Swarm Optimization, *Soft Computing*, vol. 13, issue 3, pp. 259–272, 2009.
- [24] Holden, N., Freitas, A.A.: A hybrid PSO/ACO Algorithm for Classification, *9th Annual Conference Companion on Genetic and Evolutionary Computation, ACM*, pp. 2745–2750, 2007.
- [25] Guo, Y.Z., Li, M., Lu, M., Wen, Z., Wang, K., Li, G., Wu, J.: Classifying G Protein-Coupled Receptors and Nuclear Receptors on the Basis of Protein Power Spectrum from Fast Fourier Transform, *Amino Acids*, vol. 30, issue 4, pp. 397–402, 2006.
- [26] Lundstrom, K.H., Chiu, M.L. (Eds.): *G Protein-Coupled Receptors in Drug Discovery*, Taylor & Francis Group, 2006.
- [27] Maurice, P., Guillaume, J.L., Benleulmi-Chaachoua, A., Daulat, A.M., Kamal, M., Jockers, R.: GPCR-interacting Proteins, Major Players of GPCR Function, *Advances in pharmacology*, vol. 62, pp. 349–379, 2011.
- [28] Suwa, M., Sugihara, M., Ono, Y.: Functional and Structural Overview of G-Protein-Coupled Receptors Comprehensively obtained from Genome Sequences, *Pharmaceuticals*, vol. 4, issue 4, pp. 652–664, 2011.
- [29] Dastmalchi, S., Church, W.B., Morris, M.B.: Modelling the Structures of G Protein-Coupled Receptors aided by Three-dimensional Validation, *BMC bioinformatics*, vol. 9 (Suppl 1), S14, 2008.
- [30] Alem, K.: Contribution à la Prédiction de la Structure Secondaire et Tertiaire des Protéines : Cas des Récepteurs Couplés aux Protéines G (Rhodopsin-like) du Génome Humain, thèse, Université Badji Mokhtar – Annaba, 2011.
- [31] Gloriam, D.E.: GPCRDB: GPCR data, diagrams and tools, *NC-IUPHAR news*, No 6, pp. 3, 2015.
- [32] Isberg, V., Vroling, B., van der Kant, R., Li, K., Vriend, G., Gloriam, D.: GPCRDB: An Information System for G Protein-Coupled Receptors, *Nucleic Acids Research*, vol. 42, Database issue, 2013.
- [33] Vroling, B., Sanders, M., Baakman, C., Borrmann, A., Verhoeven, S., Klomp, J., Oliveira, L., de Vlieg, J., Vriend, G.: GPCRDB: Information System for G Protein-Coupled Receptors, *Nucleic Acids Research*, vol. 39 (Database issue), pp. D309–D319, 2011.
- [34] Huang, Y., Cai, J., Ji, L., Li, Y.: Classifying G-Protein Coupled Receptors with Bagging Classification Tree, *Computational biology and chemistry*, vol. 28, issue 4, pp. 275–280, 2004.
- [35] Zekri, M., Alem, K., Souici-Meslati, L.: Identification Methods of G Protein-Coupled Receptors, *International Journal of Knowledge Discovery in Bioinformatics*, Vol. 2, No. 4, pp. 35–52, 2011.
- [36] Zekri, M., Alem, K., Souici Meslati, L. : Méthodes d'identification de récepteurs couplés aux Protéines G: Un état de l'art, *Conférence Internationale Maghrébine sur l'Extraction et la Gestion de Connaissances (EGC-M'2012)*, pp. 83–88, Hammamet, Tunisie, 2012.
- [37] Davies, M.N., Secker, A., Freitas, A.A., Mendao, M., Timmis, J., Flower, D.R.: On the Hierarchical Classification of G Protein-Coupled Receptors, *Bioinformatics*, vol. 23, issue 23, pp. 3113–3118, 2007.
- [38] Peng, Z.L., Yang, J.Y., Chen, X.: An Improved Classification of G-Protein-Coupled Receptors using Sequence-Derived Features, *BioMed Central Bioinformatics*, vol. 11, 2010.
- [39] Chou, K.C.: Prediction of G-protein-coupled receptor classes, *Journal of Proteome Research*, vol. 4, issue 4, pp. 1413–1418, 2005.
- [40] Xiao, X., Wang, P., Chou, K.C.: GPCR-CA: A Cellular Automaton Image Approach for Predicting G Protein-Coupled Receptor Functional Classes, *Journal of Computational Chemistry*, vol. 30, pp. 1414–1423, 2009.
- [41] Wheeler, D.L., Barrett, T., Benson, D.A., Bryant, S.H., Canese, K., Chetvernin, V., Church, D.M., DiCuccio, M., Edgar, R., Federhen, S., Geer L.Y., Kapustin Y., Khovayko O., Landsman D., Lipman D.J., Madden T.L., Maglott D.R., Ostell, J., Miller V., Pruitt K.D., Schuler G.D., Sequeira E., Sherry S.T., Sirotkin K., Souvorov, A., Starchenko G., Tatusov R.L., Tatusova, T.A., Wagner, L., Yaschenko, E.: Database Resources of the National Center for Biotechnology Information, *Nucleic Acids Research*, vol 35, D5–D12, 2007.
- [42] Chou, K.C.: Prediction of Protein Cellular Attributes using Pseudo-Amino Acid Composition, *Proteins*, vol. 43, pp. 246–255, 2001.
- [43] Chou, K.C.: Pseudo Amino Acid Composition and its Applications in Bioinformatics, Proteomics and System Biology, *Current Proteomics*, vol. 6, pp. 262–27, 2009.
- [44] Chou, K.C.: Using Amphiphilic Pseudo Amino Acid Composition to Predict Enzyme Subfamily Classes, *Bioinformatics*, vol. 21, issue 1, pp. 10–19, 2005.
- [45] Shen, H.B., Chou, K.C.: PseAAC: A Flexible Web Server for Generating Various Kinds of Protein Pseudo Amino Acid Composition, *Analytical Biochemistry*, vol. 373, issue 2, pp. 386–388, 2008.

- [46] Du, P., Wang, X., Chao X., Gao, Y.: PseAAC-Builder: A Cross-platform Stand-alone Program for Generating Various Special Chou's Pseudo-Amino Acid Compositions, *Analytical Biochemistry*, vol. 425, pp. 117–119, 2012.
- [47] Cao, D.S., Xu, Q. S., Liang, Y.Z.: Propy: A Tool to Generate Various Modes of Chou's PseAAC, *Bioinformatics*, vol. 29, issue 7, pp. 960–962, 2013.
- [48] Zekri, M., Alem, K., Souici-Meslati, L.: Immunological Computation for Protein Function Prediction, accepted for publication in *Fundamenta Informaticae* journal, Vol. 139, pp. 91-114, 2015.