

La sélection de paramètres d'un système industriel par les colonies de fourmis

Ouahab Kadri, L. Hayet Mouss, M. Djamel Mouss
Laboratoire d'Automatique et Productique
Université de Batna, Algérie

RÉSUMÉ: Dans cet article, nous présentons un nouvel algorithme pour réduire la dimension de vecteur d'état de fonctionnement d'un système industriel. Notre algorithme permet de sélectionner un sous-ensemble de paramètres qui offre une détection plus rapide de dysfonctionnement et une bonne qualité de classification. Cet algorithme est basé sur le comportement observé chez les fourmis réelles. Nous montrons ici que l'émergence des déplacements et les interactions des fourmis permet de trouver un ensemble réduit de paramètres qui caractérisent le fonctionnement d'un système industriel dynamique et complexe. L'algorithme offre aussi la possibilité d'utiliser des bases de données de grandes tailles. Les expériences effectuées sur les bases de données Iris et Vehicle montrent que notre algorithme fournit de très bons résultats.

MOTS-CLES: Colonie de fourmis, Classification, Diagnostic industriel, Sélection de paramètres, Système complexe et dynamique.

Introduction

Les systèmes complexes et dynamiques possèdent un comportement non linéaire et non stable durant leurs exécutions. Il est très difficile voir impossible de modéliser ce type de système. Cette difficulté est causée par le grand nombre des variables qui caractérisent l'état de fonctionnement du système.

Notre travail s'inscrit dans la surveillance de ce type de système et plus précisément dans la phase de diagnostic (figure 1). La fonction principale de la surveillance est de vérifier l'état de fonctionnement du

système. Elle est composée de deux parties qui sont la détection et le diagnostic. La phase de détection permet de déterminer l'état du système comme étant normal ou anormal. La phase de diagnostic consiste à identifier les composants défectueux et de trouver les causes à partir d'un ensemble de symptômes observés.

D'un point de vue général, la détermination des causes à partir des effets est un problème de classification. Puisque, il s'agit de comparer un vecteur de données qui représente l'état actuel du système avec des vecteurs références qui représentent l'historique de fonctionnement du système [MM08].

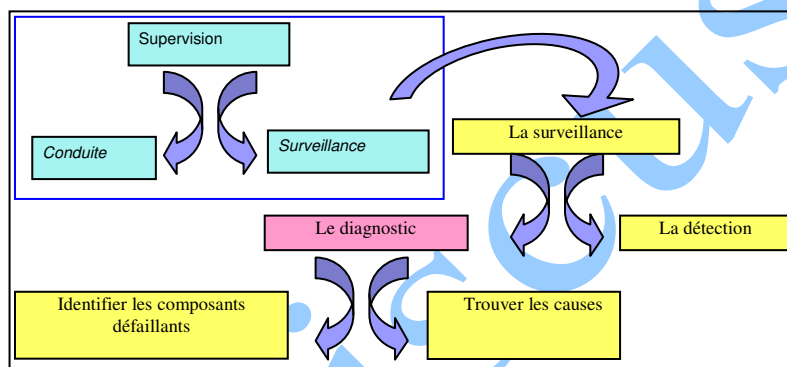


Fig. 1: Position de diagnostic par rapport à la supervision

Le diagnostic d'un système complexe et dynamique en utilisant une méthode de classification, cherche à identifier d'une façon permanente l'état de fonctionnement du système.

L'efficacité d'un système de diagnostic dépend de la pertinence des paramètres choisis. Il est donc indispensable de trouver les paramètres qui permettent de distinguer les différents modes de fonctionnement et de dysfonctionnement du système (figure 2).

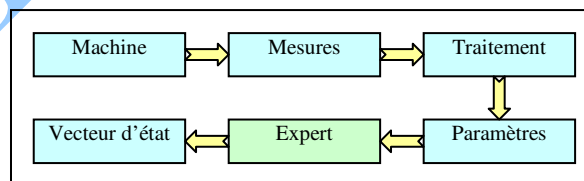


Fig. 2: Construction de l'ensemble de paramètres

L'utilisation d'un nombre énorme de paramètres influe sur la vitesse d'identification et il est possible que plusieurs paramètres n'aient pas

d'importances pour faire la distinction entre les différents états de fonctionnement.

Une méthode doit donc être utilisée afin de minimiser la dimension du vecteur d'état et ne conserver que les paramètres jugés pertinents.

Il existe deux types de méthodes de réduction de dimension : le premier type consiste à définir un nouveau ensemble de paramètres. Ces paramètres sont des combinaisons linéaires des anciens. Ces méthodes sont appelées les méthodes d'extraction. Le deuxième type consiste à sélectionner un sous ensemble de paramètres depuis l'ensemble initial. Le nouvel ensemble doit offrir une minimisation de la variance intra-classe et une maximisation de la variance inter-classe. Ces méthodes sont appelées les méthodes de sélection.

Toute méthode de sélection de paramètres consiste en quatre points essentiels :

Un ensemble de départ qui représente l'ensemble de paramètres initialement utilisés par une procédure de recherche. Cet ensemble peut être vide, ou contenir tous les paramètres ou bien un sous ensemble aléatoire.

La procédure de recherche est l'élément essentiel de toute méthode de sélection. Elle retourne comme résultat le sous ensemble de paramètres qui répondent mieux au critère de qualité. Ce critère est retourné par une fonction d'évaluation. Cette fonction détermine la qualité de classification obtenue en utilisant un sous ensemble de paramètre. Un critère d'arrêt est utilisé pour terminer la procédure de recherche. Ce critère dépend soit de la fonction d'évaluation ou bien aux paramètres de configuration qui sont définis par l'utilisateur [BL97].

La suite de l'article est organisée comme suit : la première section résume les travaux concernant la sélection de paramètres par les heuristiques. Ensuite nous allons décrire l'algorithme de colonie de fourmis ainsi nos contributions. L'étude expérimentale effectuée sur des données de la base Iris et une évaluation des performances de notre algorithme en utilisant la méthode de classification évolutionnaire ECMC [PB06] seront décrits par la suite. Et finalement on termine par une conclusion et perspectives.

1. Travaux existants

La sélection de m attributs parmi n représente un problème d'explosion combinatoire. Il existe plusieurs approches permettant d'éviter une recherche exhaustive. Parmi les premiers travaux dans ce domaine ont été

ceux de Narendra et Fukunaga [NF77], ils ont utilisé l'algorithme branch and bound qui exige la monotonie du critère d'évaluation employé. Cela veut dire que l'ajout d'un nouveau paramètre au sous-ensemble actuel ne doit pas décroître la valeur du critère. Mais en réalité, cette condition n'est pas vérifiée par la plupart des critères utilisés dans le contexte de sélection de variables. Malgré les différentes variantes proposées à cet algorithme, ils ne sont pas applicable devant les problèmes où le nombre des paramètres est très élevé.

Il existe une autre catégorie d'algorithmes est appelée Wrappers. Les deux célèbres algorithmes de cette catégorie sont Sequential Forward Selection (SFS) et Sequential Backward Selection (SBS) [PLD05]. Le point de départ de ces algorithmes est respectivement un ensemble vide ou complet de paramètres. Initialement, tous les paramètres sont évalués individuellement et on sélectionne celui qui conduit à la meilleure performance de classification. On répète cet itération jusqu'à la qualité de classification n'augmente pas. La différence entre les deux algorithmes est dans le premier, on ajoute un paramètre à chaque itération par contre on supprime un paramètre dans le deuxième [Kit78]. L'inconvénient major de cette catégorie est la forte dépendance entre le résultat final et l'ordre initial de paramètres.

Yang et Honavar [YH98] ont appliqués les algorithmes génétiques pour la sélection de paramètres. La population initiale est générée aléatoirement et elle représente un ensemble des solutions possibles. Le fitness est utilisé comme un critère d'évaluation. Le passage d'une génération à une autre est réalisé en appliquant la sélection, le croisement et la mutation. Le but de ces transformations est de converger vers une solution optimale. Les algorithmes génétiques ont donnés des résultats plus intéressants que les méthodes précédentes [R+00] et [ZCZ09].

Nous proposons dans cet article une nouvelle méthode de sélection basée sur une variante de l'algorithme de colonie de fourmis ACO [MDC96]. Nous montrons par la suite que les résultats obtenus par cette approche sont comparables aux résultats obtenus en appliquant les algorithmes génétiques.

2. L'algorithme d'optimisation par les colonies de fourmis

A. L'intelligence de fourmis

Nous remarquons chez les fourmis que la capacité observée au niveau de la colonie dépasse la somme des capacités individuelles. Les éthologistes ont montré que les fourmis sont capables de trier divers éléments du couvain, trouver le plus court chemin entre la nourriture et le nid et même de s'auto-assembler en créant des structures vivantes [K+10]. Ces phénomènes sont le résultat d'émergence des comportements simples de chaque membre de la colonie et la communication chimique entre tous les individus. Le phénomène représente un moyen efficace de partage d'information. Plusieurs algorithmes sont développés en se basant sur ce principe. Dans la suite, nous présentons le principe de l'algorithme de colonie de fourmis binaire [WCL06] et [Ani05].

B. L'algorithme binaire de colonie de fourmis

L'hybridation d'algorithmes à base de fourmis avec les méthodes génétique a donné un type d'algorithmes plus puissant et plus rapide. Ce type combine l'aspect coopératif d'algorithmes à base de fourmis avec l'aspect évolutif de méthodes génétiques. L'algorithme binaire de colonie de fourmis appartient à ce type d'algorithme. Il consiste de quatre opérations de base : croisement, sélection, mutation et la mise à jour de phéromone. Le principe général de L'algorithme binaire de colonie de fourmis peut être résumé de la façon suivante.

Génération aléatoire de la population initiale
Exécution de l'algorithme génétique n fois pour obtenir une solution qui va être utilisé pour initialiser le phéromone sur le réseau.

Répéter

- Lancer les fourmis de recherche
- Exécuter l'algorithme génétique en utilisant la solution générée par l'algorithme de colonie de fourmis
- Mis à jour de phéromone sur le réseau par la solution optimale

Jusqu'à satisfaire le critère d'arrêt

Fig. 3: L'algorithme binaire de colonie de fourmis

Le critère d'arrêt représente la dimension du vecteur obtenu par l'algorithme où le critère de qualité n'évolue pas si on ajoute un autre paramètre [Ani05].

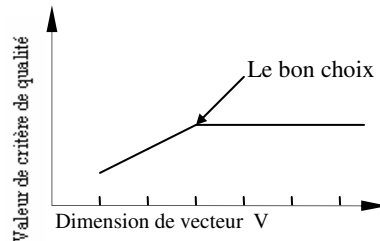


Fig. 4: Le choix de la dimension de vecteur V

Pour la représentation de l'information, nous avons utilisé un codage binaire. La dimension de chaque individu égale au nombre total de paramètres. La forme générale d'un vecteur est: $v = (x_1, x_2, \dots, x_n)$. Sachant que si le paramètre est présent, x_i prend la valeur 1 sinon il prend la valeur 0. De cette façon, uniquement les paramètres qui correspondent leurs positions à la valeur 1 seront considérés pour calculer la fonction d'adaptation (voir la figure suivante).

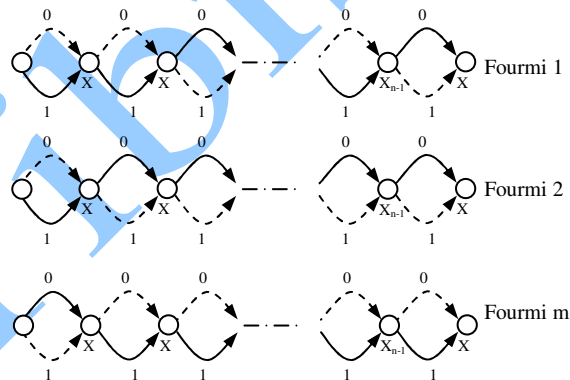


Fig. 5: L'ensemble de paramètres sélectionnés par l'algorithme Binaire

3. La Sélection par les fourmis

Nous avons proposé un ensemble de modification sur l'algorithme binaire pour l'adapter au problème de sélection de paramètres. Initialement, nous créons m agents qui représentent la population initiale. Chaque agent est

caractérisé par un vecteur V qui contient que des zéros et des uns. Après l'exécution des opérations (sélection, croisement et mutation), chaque vecteur V est associé à une fonction d'adaptation $F(V)$. La valeur retournée par la fonction F indique la qualité de la solution V . la fonction F se base sur l'utilisation de deux critères de qualités qui sont la variance intraclasse et la variance interclasse.

$$F(V) = \text{trace} \left(\sum_w^{-1} \cdot \sum_B \right) \quad (1)$$

Sachant que la matrice de variance intraclasse est calculée comme suit :

$$\sum_w = \frac{1}{N} \sum_{C=1}^M \sum_{V=1}^{N_C} (X_{CV} - m_C)(X_{CV} - m_C)^t \quad (1.1)$$

Alors que la matrice de la variance interclasse représente la séparation entre les différentes classes.

$$\sum_B = \frac{1}{N} \sum_{C=1}^M (m_C - m)(m_C - m)^t \quad 1.2)$$

Avec :

- m : centre de gravité général
- M : nombre de classes
- m_C : centre de gravité de la classe numéro C
- X_{CV} : le $V^{\text{ième}}$ vecteur de la classe numéro C
- N_C : nombre de vecteurs de la classe numéro C
- N : nombre total de vecteurs

Après l'exécution de l'algorithme génétique et l'obtention d'une solution initiale, les valeurs de phéromone sur les arcs sont initialisées en utilisant la solution trouvée par l'algorithme génétique. Ensuite, chaque fourmi de la colonie trouve une solution. L'arc choisi par une fourmi est dépend d'une probabilité PS . Cette probabilité est définie proportionnellement à un facteur phéromonal FP et un facteur heuristique FH .

$$PS_{i1} = \frac{FP_{i1} + \frac{FP_{i0}}{\text{Max}(FH)} FH_i}{FP_{i1} + FP_{i0}} \quad (2)$$

Le facteur phéromonal dépend de la stratégie phéromonale choisie et est décrit plus tard. Le facteur heuristique FH est basé sur le rapport de Fisher qui nous permet de classer les paramètres par ordre de pertinence. Ce rapport est exprimé par la formule suivante :

$$FH(\alpha) = \sum_{c=1}^M \sum_{r \neq c}^{M-1} \frac{m_c(\alpha) - m_r(\alpha)}{N_c \sigma_c^2(\alpha) - N_r \sigma_r^2(\alpha)} \quad (3)$$

Sachant que :

M représente le nombre de classes.

$m_c(\alpha)$ représente le centre de gravité de la classe numéro c en ne considérant que le paramètre α . Il est calculé comme suit :

$$m_c(\alpha) = \frac{1}{N_c} \sum_{v=1}^{N_c} X_{cv}(\alpha) \quad (4)$$

Avec X_{cv} est le vecteur numéro v de la classe numéro c . La valeur de N égale au nombre de vecteurs de la classe en question.

$\sigma_r^2(\alpha)$ est la variance de la composante α des vecteurs de la classe numéro c .

$$\sigma_r^2(\alpha) = \frac{1}{N_c} \sum_{v=1}^{N_c} [X_{cv}(\alpha) - m_c(\alpha)]^2 \quad (5)$$

Le facteur phéromonal FP représente la quantité de phéromone disposée par la colonie sur un arc. La mise à jour de FP est effectuée après que toutes les fourmis ont trouvé des solutions. Nous modifions les quantités de phéromone FP sur les arcs uniquement pour les chemins de fourmis qui ont réalisé un rapport important de la dispersion entre classes et de la dispersion intra-classe, selon la formule :

$$\Delta FP = \frac{1}{1 + F(V) - F(V')} \quad (6)$$

Sachant que : $F(V)$ représente la meilleure solution construite depuis le début de l'exécution et $F(V')$ représente la meilleure solution construite pendant la dernière itération.

Donc la formule complète de mise à jour de phéromone est comme suit :

$$FP \leftarrow (1 - \rho)FP + \Delta FP \quad (7)$$

Sachant que $\rho \in]0,1[$ et représente le taux d'évaporation. Il est utilisé pour ne pas converger rapidement vers des optima locaux.

Pour simplifier la méthode, nous n'avons pas réutilisé l'algorithme génétique dans la boucle de recherche. L'algorithme s'arrête si on converge vers une solution optimale. Mais aussi, nous pouvons fixer le nombre total d'itérations à effectuer.

4. Expérimentation

A. Les données de test

Les résultats obtenus par notre méthode sont évalués en utilisant l'ensemble Iris. Iris est une base de données introduite par Fisher en 1936. Elle est constituée de 150 enregistrements qui représentent 3 classes. Chaque enregistrement est composé de 4 éléments.

Nous avons utilisé aussi la base Vehicle. Ses données ont été recueillies en 1987 par Siebert. Vehicle est constituée de 846 enregistrements qui représentent 4 classes. Chaque enregistrement est composé de 18 attributs.

L'intérêt d'utiliser Iris et Vehicle est qu'on connaît à l'avance la classe réelle de chaque information et donc on peut évaluer la qualité de sous ensemble de paramètres sélectionnées.

B. Les paramètres de l'algorithme de sélection

Comme tout autre algorithme, avant de passer à la phase de sélection. Il faut fixer quelques paramètres. Ce problème représente l'une des inconvénients des méthodes biomimétiques. Puisque les valeurs des paramètres sont liées au nombre d'individus et la distribution des données sur la plage de représentation. Le tableau suivant présente les valeurs des paramètres de notre algorithme. Ces paramètres sont fixés après l'exécution de plusieurs simulations en utilisant comme entrée un ensemble restreint de données.

Tableau 1.
Les Paramètres de l’algorithme de Selection

Paramètre	Valeur	Description
N_A	20	Nombre d’agents
F_a	0.2	Taux de comportement aléatoire
ρ	0.3	Taux d’évaporation
S_Min_FP	Min FH	Seuil minimal de phéromone
S_Max_FP	Max FH	Seuil maximal de phéromone

C. Le facteur heuristique FH

Le facteur heuristique FH est pris en considération que par les fourmis qui ont un comportement lié à la probabilité PS. Les fourmis qui ont un comportement aléatoire sont utilisées pour découvrir des nouveaux espaces de recherche. Le tableau suivant représente les valeurs du facteur heuristique FH en utilisant la base donnée Iris.

Tableau 2.
Le Facteur Heuristique FH de la Base IRIS

FH1	FH2	FH3	FH4
0.3452	1.2554	1.0071	2.4176

D’après le tableau 2, on constate que la valeur de *FH4* va favoriser les fourmis de choisir le chemin 1 qui signifie la présence du quatrième paramètre dans le sous-ensemble final.

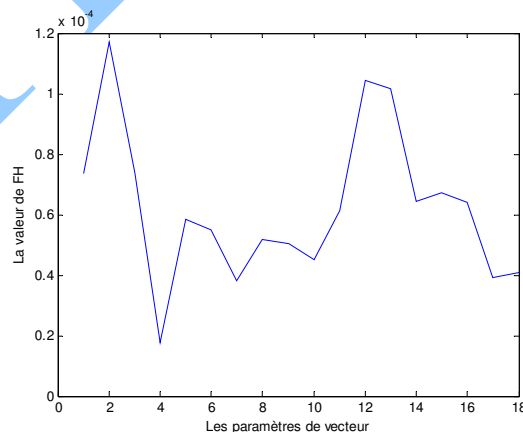


Fig. 6 : Le facteur heuristique FH de la base Vehicle

D'après la figure 6, on remarque que le deuxième paramètre possède la plus grande valeur de FH. En conséquence il va être présent dans le sous-ensemble final.

D. Les résultats

Nous avons testé les performances de notre algorithme en utilisant la méthode de classification évolutionnaire ECMC [PB06]. Le tableau suivant montre la qualité de classification en utilisant :

- Le meilleur paramètre discriminant obtenu à partir du tableau 2 et la figure 5;
- Le meilleur sous-ensemble de paramètres engendré par notre algorithme ;
- L'ensemble initial qui contient tous les paramètres.

Tableau 3.
Les Performances De Classification En Utilisant Les Différentes Entrées

	Iris Taux d'erreur	F(V)	Vehicle Taux d'erreur	F(V)
Un seul paramètre	11 %	0.0325	75 %	0.0061
Sous-ensemble généré	02 %	0.4717	10 %	0.0210
Tous les paramètres	01 %	0.6537	07 %	0.0218

Le tableau 3 montre qu'on obtient un taux d'erreur acceptable avec le sous-ensemble généré par notre algorithme. On remarque aussi que la valeur de F(V) reflète bien la qualité de classification.

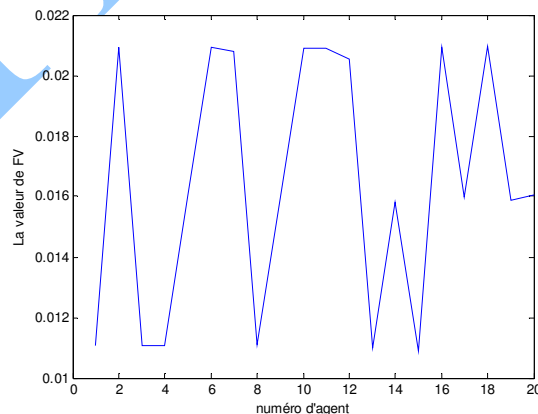


Fig. 7 : La valeur de FV obtenue par chaque agent durant la dernière itération

On remarque qu'après la dernière itération, plus de 33% d'agents arrivent à trouver la même solution optimale et cela grâce la mise à jour effectuée sur le réseau après chaque itération.

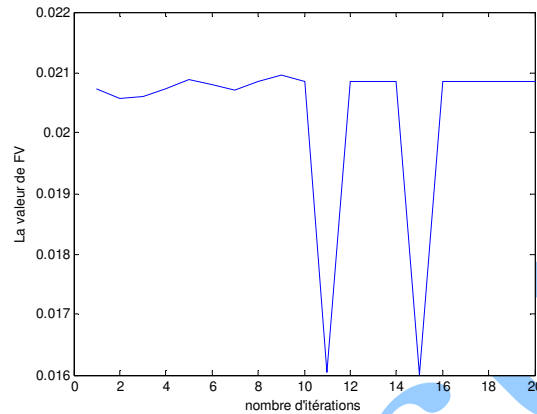


Fig. 8 : La meilleure solution obtenue durant chaque itération

La figure 8 montre qu'on obtient la solution optimale à partir la 5^{ème} itération qui montre l'efficacité et la rapidité de notre algorithme.

Conclusion

La plupart des techniques intelligentes utilisées pour le diagnostic industriel ne prennent pas en considération le problème de la taille du vecteur d'état du système. Dans cet article, nous avons présenté une nouvelle méthode de sélection de paramètre basée sur un algorithme de colonie de fourmis binaire. Notre méthode permet une sélection rapide de paramètres avec un nombre réduit d'itérations. L'originalité de notre travail est l'utilisation d'un nouveau facteur heuristique qui favorise les fourmis de choisir les paramètres les plus discriminants. Un autre avantage est le nombre d'opérations limité dans chaque itération. Les résultats obtenus montrent qu'on peut détecter l'état de fonctionnement d'un système plus rapidement et avec un taux d'erreur acceptable. Donc, il sera très intéressant d'utiliser les algorithmes de colonie de fourmis pour le diagnostic industriel.

Références

- [Ani05] A. Al-Ani - *Feature Subset Selection Using Ant Colony Optimization*, International Journal of Computational Intelligence. 2(1), pp 53-58, 2005.
- [BL97] A. L. Blum, P. Langley - *Selection of relevant features and examples in machine learning artificial*, Artificial Intelligence, 97(1-2) :245 271, 1997.
- [Kit78] J. Kittler - *Feature set search algorithms*. In C. H. Chen, editor, Pattern Recognition and Signal Processing. Sijhoff and Noordhoff, the Netherlands, 1978.
- [K+10] O. Kadri, H. Mouss, A. Abdelhadi, R. Mahdaoui - *Vers Une Optimisation De L'algorithme Anttreestoch*, International Conference On Industrial Engineering and Manufacturing ICIEM'10, May, 9-10, 2010, Batna, Algeria.
- [MDC96] V. Maniezzo, M. Dorigo, A. Coloni - *The ant system: Optimization by a colony of cooperating agents*, IEEE Transactions on Systems, Man, and Cybernetics-Part B, 26(1) :29-41, 1996.
- [MM08] R. Mahdaoui, L.H. Mouss - *Surveillance industrielle dynamique par les systèmes neuro-flous temporels: application a un systeme de production*, Conférence international en automatique bukharste romania 09/09/2008.
- [NF77] P. M. Narendra, K. Fukunaga - *A branch and bound algorithm for feature subset selection*. IEEE Transactions on Computers, C-26: 917-922, 1977.
- [PB06] M.F. Pasha, R. Budtarto - *Evolvable-NEURAL- Based Fuzzy Inference System and Its Application for Adaptive Network Anomaly Detection*. in: ICMLC 2005, LNAI 3930.PP662-671,2006.
- [PLD05] H.C. Peng, F. Long, C. Ding - *Feature selection based on mutual information: criteria of max-dependency, max-relevance,*

and min-redundancy, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 27, No. 8, pp. 1226–1238, 2005.

- [R+00] M.L. Raymer, W.F. Punch, E.D. Goodman, L.A. Kuhn, A.K. Jain - *Dimensionality Reduction Using Genetic Algorithms*, IEEE Transactions on Evolutionary Computation, A Publication of the IEEE Neural Networks Council, 4, 2000.
- [WCL06] X. Weiqing, Y.Chenyang, W.Liuyi - *Binary Ant Colony Evolutionary Algorithm*, International Journal of Information Technology. Vol. 12. 03, 2006.
- [YH98] J. H. Yang, V. Honavar - *Feature subset selection using a genetic algorithm*, IEEE Intelligent Systems, Vol. 13, n°2, pp. 44-49, 1998.
- [ZCZ09] Z. Zhousuo, W. Cheng, X. Zhou - *Research on Intelligent Diagnosis of Mechanical Fault Based on Ant Colony Algorithm*, The Sixth International Symposium on Neural Networks, Vol 56, P 631-640 Springer Berlin / Heidelberg, 2009.