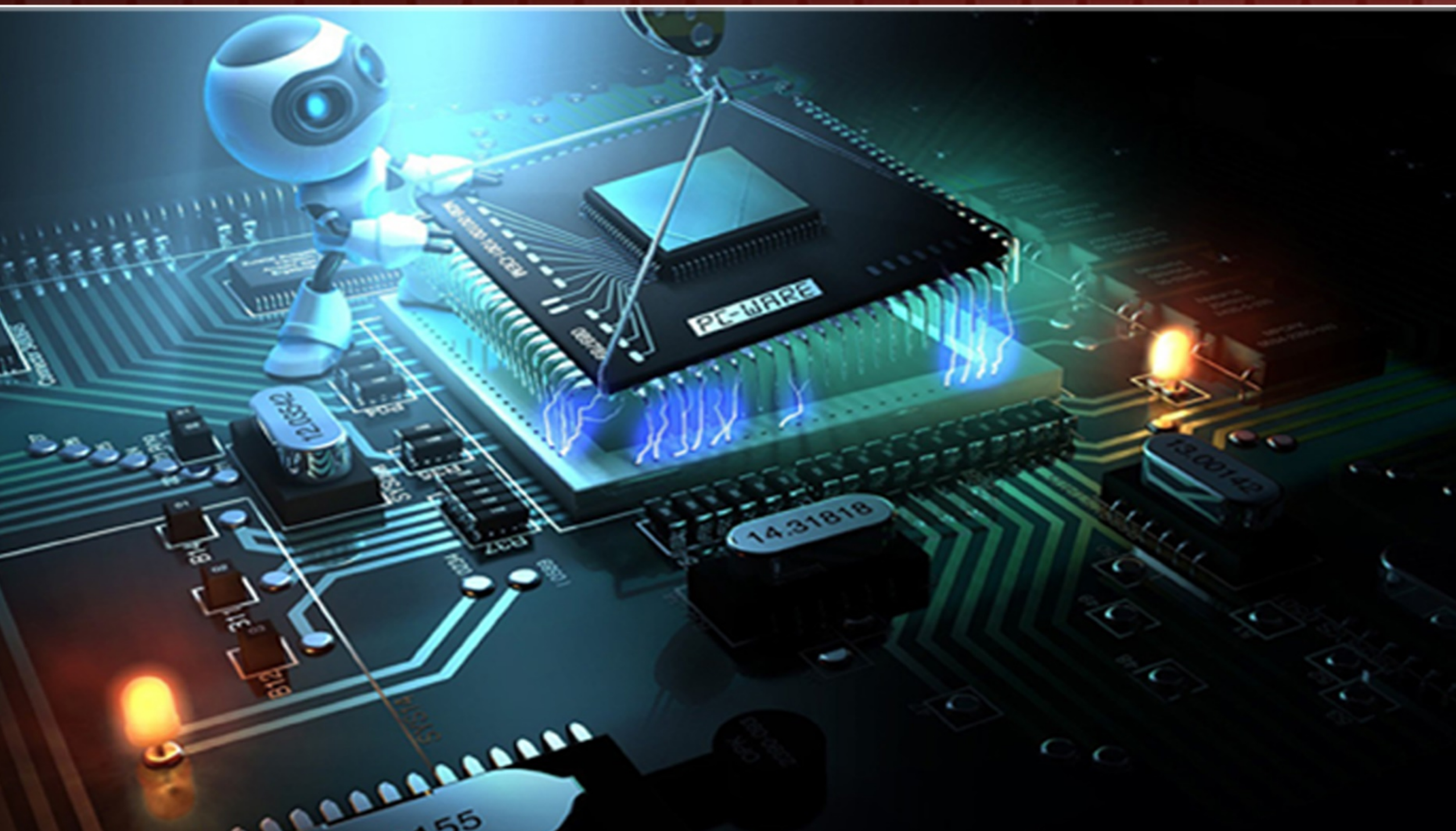
Volume 1. Issue 1

Transactions on Machine Learning and Artificial Intelligence



Society for Science and Education
UNITED KINGDOM

Table of Contents

A Survey on Application of Swarm Intelligence in Network Security

Piecewise Parabolic Approximation of Plane Curves with Restrictions in Computer-Aided Design of Road Routes

Design and Implementation of an Online Brain Computer Interface System

SOCIETY FOR SCIENCE AND EDUCATION
UNITED KINGDOM

TMLAI TRANSACTIONS ON
MACHINE LEARNING AND ARTIFICIAL INTELLIGENCE

Vol. 1 Isuue 1

A Survey on Application of Swarm Intelligence in Network Security

Muhammad Saad Iftikhar[1], Muhammad Raza Fraz[2]

[1, 2] School of Electrical Engineering and Computer Science- SEECS

National University of Science and Technology- NUST, Pakistan

[1]m.saad.iftikhar@gmail.com , [2]raza_fraz@hotmail.com

ABSTRACT

Nowadays security is an essential part of every framework. In past few years due to the increase in access of malicious data over the Internet resources the security becomes a necessary component. Swarm intelligence is an emerging and new biological field of optimization. The researchers have already developed many algorithms by studying the behavior of different swarms of incest such as Ants Bees etc. After the success of swarm intelligence in other areas researchers now started work in the field of security too. In this survey paper we tried to find out the reason of network security and how swarm intelligence method have been used to make system efficient in term of performance by providing network security..

Keywords: Network Security, Swarm Intelligence, Ant Colony Optimization (ACO), Particle Swarm Optimization (PSO), Ant Colony Clustering (ACC), Survey

1. INTRODUCTION

The use of computer and Internet is increasing day by day and they both can bring significant changes in the quality of life style of many people. In the last few years, different techniques were proposed and successfully deployed to secure the computer systems from unauthorized use. Such techniques include additional software and hardware systems like anti-virus software, firewall, secured network protocols, password protection, message encryption,

M.S. Iftikhar and M. R. Fraz; *A Survey on Application of Swarm Intelligence in Network Security* in Transactions on Machine Learning and Artificial Intelligence, Volume 1, No 1 (2013), PP 01-15
URL: http://scholarpublishing.org/index.php/TMLAI/article/view/24/TMLAI-13-1005
Publication Date: December 17th 2013

and many other security tools. With the passage of time the security methods are enhanced and hence so the methods used by attackers.

Many of the already existing networks and the new emerging technologies are expected to be secure from such Trojans and virus attacks. Systems like Mobile Ad-Hoc Networks (MANET) [1] and Wireless Sensor Networks (WSN) [2] have their own security deficiencies and vulnerabilities. It is almost impossible to make system completely secured regardless of the intrusion detection or prevention techniques. Since the already invented approaches failed to fully secure the systems the need of the new protection system or technique is essential for this purpose the intrusion detection becomes the most important part of every network for the security purposes as a second line defense after the firewalls.

In security system intrusion detection systems plays an important role in attack detection, and network inspection to protect the network or computer from any malicious data or to identify these threats. Thus such system must be efficient with respect to detection rate. After the appearance of first intrusion detection system numbers of techniques have been introduced to provide security and enhance the network performance. Few such approaches are Statistical approach, Rule Base approach, Expert System approach, Hybrid approach and Pattern Recognition technique.

Now in recent years researchers have developed interests in the field of biology and natural systems [3]. Swarm Intelligence as one of the innovative distributed paradigm that studies the behavior of swarms of insects and animals for solving complex optimization problem. Behaviors such as finding paths to food sources, organizing their nests, moving from one place to any other place in an organized way are analyzed and modeled. The security systems in networks had applied these models for the intrusion detection purpose. The purpose is to perform some significant measures such as tracing the attack source, distinguish between a normal and abnormal behavior.

Researchers are being motivated because these natural systems have many characteristics that might be used for the security purposes. Like, a swarm of insect with very confined capabilities still finishes very complex tasks. Swarm based security systems or intrusion detection systems are light in weight and simple to put into practice. They are quite robust, vastly adaptive to different conditions and more importantly self-configurable. The biology immunity mechanism has given us much reference particularly to network security. Such as intrusion detection systems, that is based on artificial immunity. The main advantage of approaching security through swarm intelligence is due to the increasing interests of swarm intelligence in the academic and industry fields. In this paper we tried to categorize the work that has been done in the field of swarm intelligence based security networks.

2. RELATED TERMS

2.1 Network Security

Network security is a generalized term used in computer networks in order to secure them from threats viruses Trojans and all the attacks that can cause damage to the computer network infrastructure. It is typically managed by some administrator a system or network administrator who can protect the whole network as well as all the recourses associated with it by implementing some security tools, software and hardware.

In computer world the word security refers to the ability of the system to protect and manage sensitive data. For example when data transmission is going between two nodes A and B over link X no one should be able to retrieve information from that link X. Network security works by applying different kind of encryption algorithms over data before transmitting it and then it is decrypted at receiving end. From the time of wireless networks till now many approaches were proposed and implemented and then replaced by better techniques or tools. These ongoing updating of security promoted the security field to be a necessary component of the network.

A network security basically relies on the layers of protections and it includes multiple components for example as a first layer defense firewalls are used while on second level defense intrusion detection systems are used. All the components work simultaneously in order to increase the overall security of the system. To make network security more effective in performance it has to identify the threats properly and combat them by choosing the best network security tool. Some common threats to network security are [4];

- Viruses, worms, and Trojan horses
- Vandals
- Spyware and adware
- Zero-day attacks, also called zero-hour attacks
- Hacker attacks
- Social Engineering
- Denial of service attacks
- Data interception
- Identity theft

As already mentioned that network security system consists of many components for the best result all the components have to work together to enhance security. These components includes [5]

2.1.1 Anti-virus software and anti-spyware software

If Anti-virus software and anti-spyware software's are updated regularly, they are able to counter most of the virus threats from the system. Antivirus and antispyware software work almost in the same way, the main difference is in the type of malicious data and pattern. The software scans the hard drive as well as the registry to detect viruses and spywares. Nowadays much antivirus software has anti-spyware software included in it or vice versa.

2.1.2 Firewall

Firewall blocks the unwanted access in any network. By using firewall properly in computer systems one can protect the system from harms by setting some filters that block the intrusion attempts from the hackers from Internet or from any other public or private network. Firewall can keep the log that how many times system is accessed. Firewall is a security device that can either be software or hardware implemented to any network. Firewall can perform many other functions too.

2.1.3 Intrusion Detection Systems (IDS)

IDS are used to monitor all the ongoing events in a network and analyze any incident which is against the network security or is a threat to network security. A more general categorization is on the basis of adopted data analysis technique. In this technique, IDS may be of the two main types. One is misuse detection and the other is anomaly detection. The misuse detection examines the whole infrastructure from attacks whereas; anomaly detection examines the protected system after some time to define what activities are normal. Any incident that considerably deviates from that kind of activities is considered an attack. It is important that IDS must detect intrusions with very high accuracy.

2.1.4 Intrusion Prevention System (IPS)

IPS identifies threats like zero hour attack, it has all the capabilities that an intrusion detection system (IDS) is and also it has the ability to stop the possible incident. The IPS has the ability to stop the attack by terminating the network connection, block the access of the unauthorized user, block all access to the targeted host and block the recourses. The IPS could also change the security environment by changing the configuration of their security controls to stop an attack. Few technologies can remove or replace malicious part of an attack thus IPS may also change the attack contents.

2.1.5 Virtual Private Network (VPNs)

VPNs provide secure access and data encryption between two peers on a network. Through VPN remote users can access the network without being hacked or intercepting data. For example it allows you to be sitting at some other place and access your company's system in the same way as if you were sitting at the company's office. It is almost impossible for the hackers to tap with data in the VPN tunnel.

2.1.6 Data Encryption and Network Security

Data encryption is used to ensure that original data is secured and cannot be intercepted by any unauthorized user. This is also known as cryptography. Cryptography is the most effective and a secure way to send a data via any public network or over Internet. In this technique we converted our original message with some encrypted code word by using algorithms. And on the other end the opposite peer must know the decrypted key so that to successfully decrypt the original message back

2.2 Swarm Intelligence

Nature always plays a vital role to solve complex human problems. In the past few years biology based techniques get the attentions of researchers. Many natural biological inspired techniques have been proposed for the network security, one of them is swarm intelligence.

Swarm intelligence is the, "The emergent collective intelligence of groups of simple agents."[6]. It is a computational intelligence approach to solve real world complex problems. It was first introduced in cellular robotics system by Beni and Wang in 1989[7]. Swarm intelligence systems buildup of a population of simple agents interactive with each other individually or with their environment. The inspiration of swarm intelligence comes from the biological or natural system. Example of SI includes ant colonies, bees, fish schooling, bird flocking and animal herding bacterial growth.

All the research, techniques, algorithms and approaches that are done up till now are after studying the behavior of swarms of insects fishes and birds. The complex problem that seems almost impossible at individual level is solved by insects, bees and birds in the form of swarms. Individually ants and bees have very limited brain and hence no intelligence but when these ants and bees interact with others to make a society then they seems to do really hard and complex tasks such as finding secure path to food, build their nests, travel in a line and synchronize their movements so that it looks like a single coherent entry with high speed etc.

This pattern becomes more significant when they doing their tasks in the absence of any centralized administrator (e.g., queen of hive). Implementation of this is seen in the NP-hard optimizations problems such as the scheduling, traveling salesman, vehicle routing etc. Researchers have done so many work in this field and create many swarm intelligence based algorithms and applications. Few important algorithms and applications are;

Algorithms:

- Ant colony optimization algorithm
- Artificial bee colony algorithm
- Particle swarm optimization
- Firefly algorithm
- Multi-swarm optimization
- Ant colony cluster optimization

Applications

- Ant-based routing
- Crowd simulation
- Swarmic art

Three main Algorithms used in network security are ACO, PSO and ACC.

2.2.1 Ant Colony Optimization Background

One of the main algorithms on swarm intelligence is ant colony optimization algorithm. It is based on the behaviors of ants. The inspiration of this idea is gain from the actions of ants and their ability to find the shortest secured path of food from that place to their nests. This is one of the most successful biologically inspired algorithms. Ants individually have no intelligence and without any vision. They are unable to communicate as they are deprived of the speech. However, their actions are strictly organized and well mannered. This shows that there is still some kind of communication between them.

Researches after conducting many experiments came to know that natural ants deposited a stuff known as pheromone traces, which direct others to the food recourses. Initially ants move randomly around in search of food. When they find their food they carry it to their nests and lay down pheromones traces so that others also know this path. Then the ants decided which path is the shortest and secured path to the food based on the pheromone concentration, thus they mostly choose the path that have greater concentration of pheromones.

Deneubourg et al. performs a double bridge experiment. In this experiment he separated the food material and ant's nests by a bridge, which has two branches of equal lengths[8]. He noticed that the one path is used by majority of the ants and this selection is very randomly. After this experiment Goss et al. extended this by adding two unequal lengths [9], all his experiments shows that majority of the ants choose the shortest path that is illustrated in Figure 1.

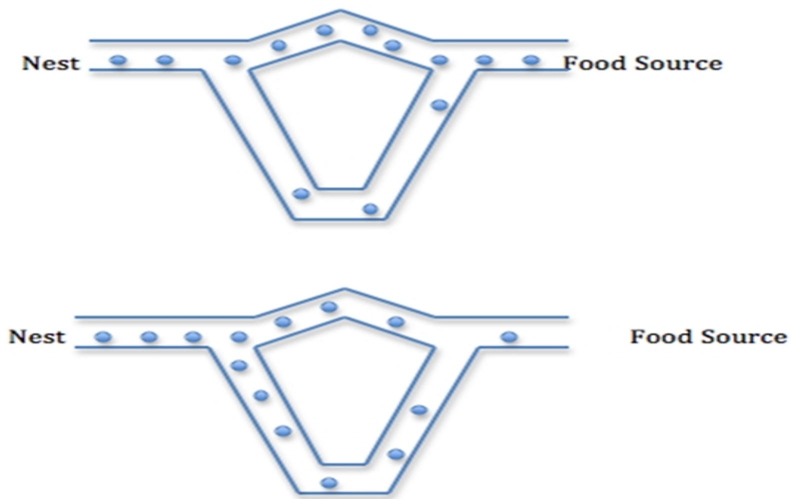


Figure 1: Extended Double Bridge Experiment

2.2.2 Particle swarm Intelligence Background

Particle swarm optimization (PSO) is computational method optimization technique which incorporate swarming behavior in swarms of bees, flocks of birds, schools of fish or also somehow from the human social behavior. PSO seeks inspiration [10] in the coordinated movement dynamics of birds and fishes etc. PSO proves that the kinesiology of the whole flock of birds is the result of any individual actions of a bird that follows 3 simple n basic rules.

i. Collision avoidance; which tell them to adjust their position in according to their mates.
ii. Velocity matching; which tell everyone to always synchronize their speed with the speed of their mates.
iii. Flock centering; which tell everyone to stay closer to their mates

In past few years researchers applied PSO successfully in many fields, and it also observed that PSO can give better and faster results compared to other techniques. The main strength of PSO as an algorithm is its fast and cheaper convergence, which is supposed to b more appropriate when, comparing it with other global algorithms like Genetic algorithm (GA), simulated annealing (SA) and other optimization algorithms.

Dr. Eberhart and Dr. Kennedy developed the basic PSO model in 1995[25]. According to this model a function exists. i-e $f : R^n \rightarrow R$ which is known as fitness function and it measures the quality of current solution. Inside the hyperspace a number S is randomly placed at $x_i \in R^n$ and have a random velocity $v_i \in R^n$ This particle move randomly in the hyperspace and after every step its position is evaluated according to the fitness function and speed updated by

$$v_i(t+1) = wv_i(t) + c_1 r_1 (p_i - x_i) + c_2 r_2 (g - x_i)$$

Where in above equation w denotes the constant inertia weight, c_1 and c_2 denotes the acceleration constants, r_1 and r_2 are random numbers, p_i is the personal best position of particle *i*, *g* is the global best position among all particles in the swarm, and x_i is the current position of particle *i*. Furthermore, the equation for the new position is:

$$x_i(t+1) = x_i + v_i(t+1)$$

The basic features of PSO model are (a) Speed and the next position of each particle is calculated according to their position of that individual with respect to the others and (b) the best solution is communicated to every other individual in the swarm. It is therefore clear that this algorithm is quite easy to implement and also researchers seeks inspiration because of the similarities between PSO and other generic algorithms. Last but not the least, PSO keep a kind of memory, which is necessary to the convergence to an optimal result.

2.2.3 Ant Colony Clustering

In optimization research area the ant colony clustering technique which is the extension of ant colony optimization algorithm. Many social insects such as ants and bees behaves in an amazing manner in the matter of organizing their nests arranging foods and cleaning of garbage. Research has proved that such insects have very high swarm intelligence. However, they are strict to some families of similar objects. Moreover it is observed that the highly intelligent is ants because of some their work and behavior.

One such example is they work in a group and if some external enemy attacks their nests they have the ability to reconstruct it in no time. And it has been observed that they worked voluntarily without being ordered by some administrator. On this observation many mathematical models are proposed and construct to create the cataloging and clustering behavior of the real ants. Deneubourg et al. was the first to construct a basic mathematical model on the basis of this behavior and this model is then used in robotics[11].

3. SWARM INTELLIGENCE IN NETWORK SECURITY

Sometimes to solve big network security problems we have to think something on a small scale, for example of an ant. One of the researches at Pacific Northwest National Laboratory is to resolve the computer-generated security problems. , Dr. Glenn Fink, one of the senior research scientists at PNNL believes that Nature always shows us paths that how can we protect networks specially computers with the help of collective intelligence. Dr. Glen together with Dr. Errin Fulp (Associate Professor of Computer Science at Wake Forest University,) works on the security software. Both the researchers studied the behavior of ant and are able to develop multiple security scanning softwares that is capable of scanning different threats.

Dr. Fulp explains the reason why they chose to copy ant's behavior in the article "Ants vs. worms" by Eric Frazier at wake Forest University.

> *"In nature, we know that ants defend against threats very successfully. They can ramp up their defense rapidly, and then resume routine behavior quickly after an intruder has been stopped. We are trying to achieve that same framework in a computer system."[12]*

To understand their behavior researchers watch National geographic documentaries about the ants and their colonies. They name their technology as swarm intelligence because according to definition the SI system, It is a system made from the population of birds, ants and agents interacting on simple rules with one another and also with their mates.

The researchers propose a digital swarm intelligence system. This SI system's body consists of three main parts.[13]

1) Digital Ant

These are the software defined ants and have the ability to crawl in the computer code and see for the malicious data or indication of the malware. Researchers mentioned that almost 3000 different types of digital ants have to be engaged.

2) Sentinel

Sentinel is the autonomic administrator or manager for digital ants on a single computer. The purpose of the sentinel is to collect information from the digital ants and check the status of the local host. On the basis of this information it decides whether to take action or not. Furthermore, sentinel also reports to sergeant.

3) Sergeant

Like sentinel, sergeant is also an autonomic controller. But as sentinel is for digital ants, sergeant is for multiple sentinels. Number of sergeants in any network is direct with the network size, size of the network determine the numbers of sergeant for that network. It is the interface for human administrator or manager.

3.1 Working of Digital ANT

Working procedure of digital ant is slightly different from that of typical antivirus scanners. Different digital ants crawl in the system randomly in search of any malware. They are simply checking network statistics or programs info, process info table hence different ant population check for different areas in the same computer. When any of these agents (ANT) found something abnormal they leave their pheromone trail immediately. This attracts other ants towards that section. As the different ants are checking at different areas they start crawling towards that abnormal area. The sentinel monitors this movement and then takes action as it is trained to understand the “normal” behavior. And then sergeant approaches to sentinel and check for the changes and threats.

If the working digital ant is helpful in finding malware threats and information then more ants might be created by simple duplication. While on the other hand if it is useless it dies automatically. The population is maintained in either case. The informative digital ant is live as long as it is supplied by the reward in the form of energy. If it is unsuccessful the energy level decreases and the ant terminates. The ideas of collective intelligence of anti-virus developers is almost similar to that of swarm intelligence, the difference resides in the number of working agents.

3.2 Intrusion Detection Systems

Security in networks is of various types. It may be of first level defense systems such as antivirus and Trojans or might be second level defense systems that are intrusion detection intrusion prevention systems. In swarm intelligence there are number of approaches researchers found based on ant colony optimization, particle swarm optimization and ant colony clustering intrusion detection and prevention systems. Some of the work of different researchers in this area is briefly described below.

3.2.1 IDS approaches based on Ant Colony Optimization

As describe earlier that IDS are used to identify the worms in the networks. Fenet and Hassas are the first to propose the intrusion detection systems based on the ant colony optimization that can detect the source of attack in the system [14]. Their proposed system is based on the number of agents. They used mobile as well as static agents for this purpose. One of the necessary component, pheromone is a static server which is available at each host and is protected. One of the duties assigned to pheromone server is that it alerts the network whenever any attack is observed. This alert message is received as a pheromone of the ant. Another static agent is *watcher, which* is also available at each host and is monitoring all the processes on the host side as and its network links. Thus watcher is supposed to be the main part of the system for detection purpose. Now as far as the mobile agents are concerned *lymphocytes* are included in this part. They are always in motion and wandering different parts of the system in search of a pheromone.

The lymphocytes are mobile agents that typically roam randomly through the network searching for pheromone marks. If they found pheromone somewhere they attract towards that area and alarm the system to take action against that threat. Since the *lymphocytes* are the responsive component of the system thus this ant colony analogy is a responsive part of the systems so that intrusions can be detected effectively and fast. The complete system forms a fully distributed ID&R system. Another approach that identifies the attacks and responds to the attacks is IDReAM by[15]. In this approach the detection part is done by the inspiration of human immune system and the intrusion response system is same it works on the ant colony optimization approach.

Abadi and jalali were the first to introduced ANTNag algorithmic approach based on ant colony optimization technique for intrusion detection[16]. The idea behind their motivation was that the systems are susceptible. Intruders set free their attacks and all the possible attacks are represented by a graph called Network Attack Graph (NAG). NAG is a directed graph. The edges and paths nodes are the representation of complete attack scenario. Then with the information on these NAGs numbers of ants are created and incremented until all the area under attacked covered. For the accurate and effective results one has to analyze the susceptibility of the system accurately. However in realistic world NAG generation is a complex and difficult process.

ACO is implemented as a detection source for an attack or it also defines rules that which behavior is normal and which one is considered as abnormal.

One of another outstanding work in the field of security particularly in intrusion detection on the basis of ACO was by Soroush in 2006 [17]. He proposed a classification system based on ANT-Miner [18]. This system is inspired by the work of Parpinelli's Ant-Miner rule extracting algorithm. The main difference of this classification is that it involves number of ant colonies instead of a single colony like other approaches of Ant-Miner. Ants are described in classes say class A and class B, then it is noticed that if some ants from class A, lost their pheromones trail and mixed with the searching of class B ants than this algorithm is put back and this situation is resolved by putting same kinds of ants in same colony. Such as if any ant leaves pheromone than only the member of that particular colony is attracted towards it. By experimenting on different colonies researcher find at last one rule for one colony. And at the end the best quality rules are added to the rules set.

3.2.2 IDS approaches based on Particle Swarm Optimization

Like ACO work has been done in the field of network security on particle swarm optimization techniques. Dozier et al. first proposed a PSO based system. The purpose of this system is to detect those threats or attacks that would skipped by the system's security and considered as a normal traffic [19, 20]. Majority of the PSO based IDS systems are hybrid in nature and are categorized in addition to the maximum-likelihood approaches. Some of the Hybrid PSO methods are described below.

3.2.2.1 PSO based neural hybrid methods

Researchers worked out number of application based on artificial neural network for intrusion detection purpose. Artificial neural network is supposed to be the major approach of soft computing for data classification. Artificial neural network combined with Particle swarm

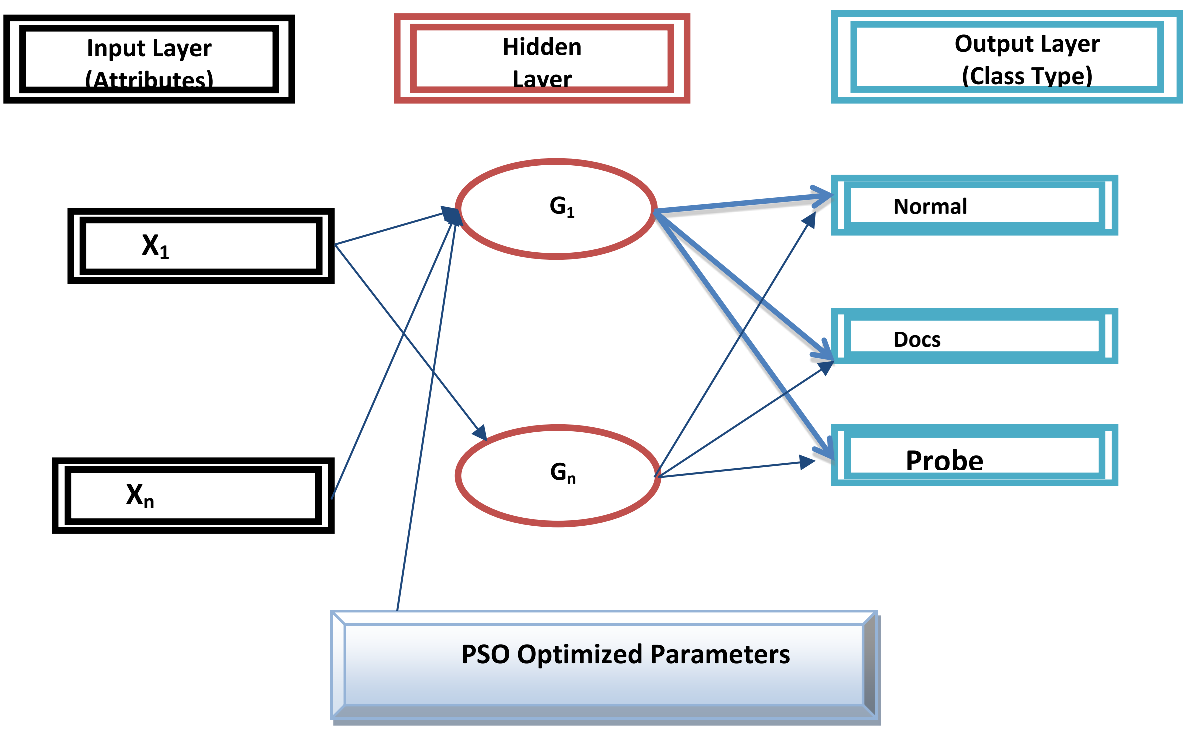


Figure 2: ANN Combined With PSO

optimization is used widely in the field of computer security and intrusion detection. The one who combines the above-mentioned soft computing methods to form an effective intrusion detection [21]. They proposed an integrated intrusion detection system, which is then implemented in java. In their training duration they use PSO recursively to get the optimal results. Each element in the hybrid PSO relates to the synaptic weight (potency and amplitude) of the corresponding network. The best result synaptic weight is then sent to artificial neural network part, which then performs the classification with better effectiveness. Their system is consisting of two main parts. (a) Artificial neural network classifier, it has to complete the classification process, (b) PSO algorithm, which is running on above to train the synaptic weight and improve other important parameters, which can be easily depicted from Figure 2.

3.2.2.2 Hybrid PSO and K-Means Algorithm Methods

Xiao [22] integrate the K-Means algorithm [23] with the particle swarm optimization technique to build an effective intrusion detection system. Yongzhong also presents same kind of K-Means hybrid system with the combination of PSO [24]. This algorithm tells that position of the each element is a set of D dimensional centroids that is generated by the K-Means algorithm. Position of corresponding element is then represented by an array.

$$\begin{pmatrix} Z_{11} & \rightleftharpoons & Z_{1D} \\ \vdots & \ddots & \vdots \\ Z_{k1} & \cdots & Z_{kD} \end{pmatrix}$$

In this expression D denotes the dimensions of the centroids and k denotes the clusters. As an initial step all the data points are assigned to k number of clusters in a random way. After this assigning, centroid is calculated and each particle's position is analyzed.

As mentioned above in the PSO section that PSO works on the basis of position and velocity thus for each element in the array , the fitness function calculates the corresponding position and velocity of the element and also update the P_{best} and G_{best} values of the particle. And in the end this algorithm works to optimize the new production of particles. The advantage of his algorithm is that it converges to local optimum with very low probability and it has very high convergence speed as compared to other algorithms.

4. CONCLUSION

Security is one the most important concern of any network whether you talk of the local Area Networks (LANs), Wide Area Networks (WANs), Metropolitan Area Networks (MANs) ,Wireless Sensor Networks (WSNs) [2]or mobile & ad-hoc Networks (MANET)[1]. Thus researchers are attacked towards this field in past few years. In this survey we briefly explain few methods, which are proposed by different scientists and researchers, of swarm intelligence optimization in network security. As the network technologies invented, security concerns are also increased. Swarm intelligence is comparatively a new technique in network security.

Swarm intelligence techniques make themselves a solid alternative for any current security techniques especially when you are talking of intrusion detection systems. Ant colony optimization (ACO) an easy to implement and fast optimization algorithm helps in the detection process of the malware. Many of the existing systems are reply on this technique in combination of some other rule based techniques. While, particle swarm optimization (PSO) and Ant colony Clustering techniques are successful techniques which provide optimal detection rate (DR). ACC is excellent in all the classes except one that is (U2R). However the main purpose of this survey is finding out the applications of swarm intelligence in network security though a very important field that is neglected up till now is the creation of distributed intrusion detection systems because security threats remain major attraction for the researchers as long as there are ways to intimidate the data in network.

REFERENCES

[1]. Hao, Y., et al., Security in mobile ad hoc networks: challenges and solutions. Wireless Communications, IEEE, 2004. 11(1): p. 38-47.

[2]. Pathan, A.S.K., L. Hyung-Woo, and H. Choong Seon. Security in wireless sensor networks: issues and challenges. in Advanced Communication Technology, 2006. ICACT 2006. The 8th International Conference. 2006.

[3]. Williamson, M.M., Biologically inspired approaches to computer security. Information Infrastructure Laboratory, HP Laboratories Bristol, 2002.

[4]. Systems, C. What is network security. 2013 [cited 2013 March]; Available from: http://www.cisco.com/cisco/web/solutions/small_business/resource_center/articles/secure_my_business/what_is_network_security/index.html.

[5]. University, B. Network Security. 2010 [cited 2013 march]; Available from: http://www.cc.boun.edu.tr/network_security.html.

[6]. Martino, G., F. Cardillo, and A. Starita, A new swarm intelligence coordination model inspired by collective prey retrieval and its application to image alignment. Parallel Problem Solving from Nature-PPSN IX, 2006: p. 691-700.

[7]. Beni G, W.J., Swarm intelligence in cellular robotics systems, 1989: NATO Advanced Workshop on Robots and Biological System

[8]. Deneubourg, J.L., et al., The self-organizing exploratory pattern of the argentine ant. Journal of insect behavior, 1990. **3**(2): p. 159-168.

[9]. Goss, S., et al., Self-organized shortcuts in the Argentine ant. Naturwissenschaften, 1989. **76**(12): p. 579-581.

[10]. Reynolds, C.W. Flocks, herds and schools: A distributed behavioral model. in ACM SIGGRAPH Computer Graphics. 1987. ACM.

[11]. Kennedy, J. and R. Eberhart. Particle swarm optimization. in Neural Networks, 1995. Proceedings., IEEE International Conference on. 1995. IEEE.

[12]. Frazier, E. Ants vs Worms. 2009 [cited 2013 march]; Available from: http://www.wfu.edu/wowf/2009/20090921.ants.html.

[13]. Kassner, M. Swarm Intelligence: Are digital ants the answer to malware. 2009 [cited 2013 March]; Available from: http://www.techrepublic.com/blog/security/swarm-intelligence-are-digital-ants-the-answer-to-malware/2757.

[14]. Fenet, S. and S. Hassas, A distributed Intrusion Detection and Response System based on mobile autonomous agents using social insects communication paradigm. Electronic Notes in Theoretical Computer Science, 2002. **63**: p. 41-58.

[15]. Foukia, N. IDReAM: intrusion detection and response executed with agent mobility architecture and implementation. in Proceedings of the fourth international joint conference on Autonomous agents and multiagent systems. 2005. ACM.

[16]. Abadi, M. and S. Jalili, An ant colony optimization algorithm for network vulnerability analysis. Iran. J. Electr. Electron. Eng, 2006. **2**(3): p. 106-120.

[17]. Soroush, E., M.S. Abadeh, and J. Habibi. A Boosting Ant-Colony Optimization Algorithm for Computer Intrusion Detection. in Proceedings of the 2006 International Symposium on Frontiers in Networking with Applications (FINA 2006). 2006.

[18]. He, J. and D. Long. An improved ant-based classifier for intrusion detection. in Natural Computation, 2007. ICNC 2007. Third International Conference on. 2007. IEEE.

[19]. Dozier, G., et al., Vulnerability analysis of immunity-based intrusion detection systems using genetic and evolutionary hackers. Applied Soft Computing, 2007. **7**(2): p. 547-553.

[20]. Dozier, G., et al. Vulnerability analysis of AIS-based intrusion detection systems via genetic and particle swarm red teams. in Evolutionary Computation, 2004. CEC2004. Congress on. 2004. IEEE.

[21]. Michailidis, E., S.K. Katsikas, and E. Georgopoulos. Intrusion detection using evolutionary neural networks. in Informatics, 2008. PCI'08. Panhellenic Conference on. 2008. IEEE.

[22]. Xiao, L., Z. Shao, and G. Liu. K-means algorithm based on particle swarm optimization algorithm for anomaly intrusion detection. in Intelligent Control and Automation, 2006. WCICA 2006. The Sixth World Congress on. 2006. IEEE.

[23]. MacQueen, J. Some methods for classification and analysis of multivariate observations. in Proceedings of the fifth Berkeley symposium on mathematical statistics and probability. 1967. California, USA.

[24]. Liu, L. and Y. Liu. MQPSO based on wavelet neural network for network anomaly detection. in Wireless Communications, Networking and Mobile Computing, 2009. WiCom'09. 5th International Conference on. 2009. IEEE.

SOCIETY FOR SCIENCE AND EDUCATION
UNITED KINGDOM

TMLAI TRANSACTIONS ON MACHINE LEARNING AND ARTIFICIAL INTELLIGENCE

Vol. 1 Isuue 1

Piecewise Parabolic Approximation of Plane Curves with Restrictions in Computer-Aided Design of Road Routes

Valery I. Struchenkov

Moscow State University of Radio Engineering, Electronics and Automation, Moscow, Russia

Email: str1942@mail.ru

ABSTRACT

Approximation problems of plane curves, which are set as a sequence of points, arise in computer-aided design of roads. Approximating curve consists of the elements: straight-line and parabolas segments. The parameters of these elements are constrained. Moreover, the number of elements is unknown. This article deals with the problem of per-element approximation, in which the elements must meet to the restrictions of special kind. This problem arises in computer-aided design of the longitudinal profile of road. The problem is solved by dynamic programming.

Keywords: Approximation, restrictions, dynamic programming, nonlinear programming

1. INTRODUCTION

The route of road — is the three-dimensional curve, satisfying a number of restrictions. The traditionally required three-dimensional curve is represented by two plane curves: horizontal and vertical alignment (later: plan and profile). The plan curve is a projection of the 3D curve to the coordinate plane XOY, and the longitudinal profile is the coordinate **z** as a function of the length **s** of the curve in the plan.

The position of the route is influenced by many factors (topography of the land, geology, hydrology, climate, etc.) and a set of restrictions on the plan and longitudinal profile. Therefore, the task of finding the optimal route as a space curve is not yet formalized in adequate mathematical models.

For this reason in the current computer aided design software (CAD) the planning is carried out manually [1,2,3]. In the new system "Trimble Quantum" computer calculates and offers for designer a few tens of route variants [4] using heuristic algorithm. The presence among them a

Valery I. Struchenkov; *Piecewise Parabolic Approximation of Plane Curves with Restrictions in Computer-Aided Design of Road Routes* in Transactions on Machine Learning and Artificial Intelligence, Volume 1, No 1 (2013), PP 16-26
URL: http://scholarpublishing.org/index.php/TMLAI/article/view/10/TMLAI-13-1015
Publication Date: December 17th 2013

route, which is near to the optimum one, is not guaranteed. An alternative approach is to assign multiple plan options manually and later to project the longitudinal profile for each of them by computer. This approach has been implemented in the new CAD system [5,6]. The number of elements projected line is unknown. For this reason, the problem is solved in three steps.

1. Projected line is represented as a broken with elements of short length. It must satisfy all restrictions except the restrictions on the minimum length of the element. We believe that its nodes and nodes of the ground profile have the same abscissa. The ground profile is always represented as a broken with an uneven step, and this assumption makes it possible to fix the number of elements (the dimension of the problem) and the lengths of the elements [5,6]. The problem is solved using a nonlinear programming algorithm [6]. The cost of earthworks and engineering structures (pipe-culvert, bridges, etc.) are included in the objective function. For each structure its cost as a function of the corresponding working mark (the height of the mound) must be set before projecting. Possibility of changing the type of structure for large changes in the working marks is accounted.

   Thus, the cost model of objective function takes into account the relationship between the adjacent project tasks. Computer simultaneously with the designing of the longitudinal profile projects cross- sections, and makes a choice of types of engineering structures. The calculations are repeated in the refinement of the input data. Project line, which we obtain as result of optimization, does not satisfy the restrictions on the minimum length of the element. Possible deviations after it transformation to the final form in accordance with the current design standards do not exceed 0.6 m.

2. We must transform broken line to the sequence of parabolas with minimal deviations.
3. As a result of second stage we define the real dimension of the problem and the initial approach. At the last stage we perform optimization with all restrictions and the necessary revisions to the objective function.

At the first and third stages, we use a nonlinear programming algorithm. At the second stage, when working on low-power computers, we used a heuristic algorithm sequential selection of the elements with returns when deviations become large. Now we can solve this problem strictly using dynamic programming. The presentation of this algorithm is the purpose of this article.

2. STATEMENT OF THE PROBLEM

Approximated curve is given by a number of points in the Cartesian coordinates $L_1 = \{x_i, y_i\}$ (i = 1,2, ..., n), i.e. represented as a broken line. It is required to approximate this broken line using a smooth curve consisting of straight and parabolic elements of 2nd degree, which satisfy the linear restrictions of special type. The number of elements in a required curve is unknown and must be obtained in the process of solving the problem. All restrictions must be satisfied and quantitative criterion must be minimum. As such a criterion can be considered the maximum deviation (on absolute value) from the original curve or the integral of the square of the difference between the original and the approximating functions . The restrictions of four types must be satisfied.

1. On the differences between the approximating curve and the original broken line at specified points.
2. On the first derivative of the approximating curve at all points.
3. On the curvature of the approximating curve at all points.
4. The lengths of the elements of the approximating curve must be greater than specified values. In practical problems this restriction will be considered as a restriction on the difference between the abscissa of the end and start points of the elements.

It is considered that the start and end points of the approximating curve, and the initial and final direction are given. At the boundaries the elements have a common tangent.

3. THE METHODOLOGY

First, consider a simple and highly unlikely event that we know the number of elements and the abscissa of their ends. We want to find the parameters of the elements. For this purpose in the search area we will make the grid of variation. On the end of each element we will defer up and down relatively original line a specified number of discrete. (Figure 1). The value of the discrete is set based on the required solution accuracy and computational capabilities.

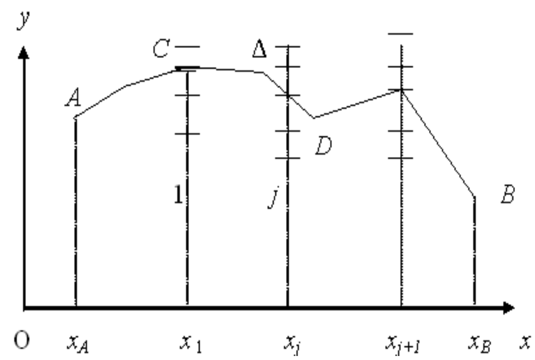


Figure 1. The grid construction. ACDB- initial line

At the first step (Figure 2) to each point of the vertical 1 goes only one parabola, because the starting point A and the initial direction are fixed.

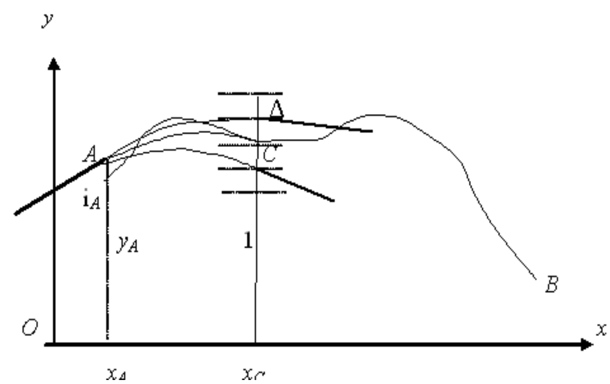


Figure 2. The variants of the first parabola

The equation of the parabola is $y = ax^2 + bx + c$. If the end ordinate of parabola y_c is given, then we have the system of three equations for determination the parameters a, b and c :

$$ax_A^2 + bx_A + c = y_A$$

$$ax_C^2 + bx_C + c = y_C$$

$$2ax_A + b = i_A$$

Here, the initial slope $i_A = tg\alpha$, where angle α sets the initial direction with the axis OX.

We find consistently a,b,c.

$$a = \frac{y_C - y_A - i_A\left(x_C - x_A\right)}{\left(x_C - x_A\right)^2}$$

$$b = i_A - 2ax_A \text{ and } c = y_A - ax_A^2 - bx_A$$

The slope (derivative) at the end of the element is equal

$$i_C = 2 (y_C - y_A)/(x_C - x_A) - i_A$$

If $(y_C - y_A)/(x_C - x_A) = i_A$, then instead of a parabola, we have a straight as special case ($i_A = i_C$). For different values of y_C we get different shapes (convex, concave, with the vertex inside or outside the element). Note, if we change y_C on Δ (discrete) then final slope i_C changes on $2\Delta/(x_C - x_A)$. If at least one parameter of some parabola doesn't satisfy at least one restriction, then such parabola is ignored. For each of the remaining parabolas we calculate the value of criteria (index of quality approximation). The restrictions on the curvature are reduced to bilateral restrictions on the parameter **a,** if the slopes (the first derivative of the approximating curve) much smaller than unity, which is the case in the problems of road design. In any case, it is easy to verify all restrictions for any variant of the parabola.

Instead of complete enumeration of options y_c, we can calculate the parameters a, b, c minimizing the integral of the square of the difference between the original and the approximating curve in the considered interval. If the restrictions are violated, we correct the value of corresponding parameter and then calculate yC. However, we must add the grid (some nodes) relatively to the resulting point, which is close to the point on the original curve with the same abscissa. So, the use of the best (local!) approximation has no meaning

When we construct the second and all subsequent elements, each point on the vertical1 should be seen as the start. On the first element that was the point A with one possible value of the initial slope, now there are a lot of points, and in each of them there are several values of the initial slope.

If the number of points on each vertical is equal m, then the number of paths that lead from the starting point A to the second vertical is equal m, to the third vertical is equal m^2, later m^3 and so on.

The part of these paths is unacceptable, but overall their number is growing rapidly. Due to restrictions we cannot take a single point on the vertical as the "system state". It is necessary to add the value of slope (derivative) in it. We may compare only paths converging with the same slopes. It means that the paths which we may compare have a common end point and end slope. Of all such paths we leave only one for further consideration. Its value of total criterion is minimal.

We denote x_j, y_j-coordinates of the beginning of the j^{th} element, $L_j = x_{j+1} - x_j$, i_j - the slope at the beginning of j^{th} element. In accordance with formula for the end slope, we may compare variants of j^{th} elements, which have the same length Lj, if $2(y_j''-y_j') = (i_j'-i_j'')L_j$. Here y_j', i_j' and y_j'', i_j'' the ordinates and the slopes at the beginning of the j^{th} element for two its variants , which have the same end point and end slope.

At such rule of rejection the variants their number increases dramatically, resulting in an insuperable computational difficulty even if the lengths of the elements are known. Instead of, we assume that variants which converge at one point are comparable if the difference of their final slopes is small, that is, we introduce the discreteness of the slopes. This means that we use two parameters for each state but the grid for second parameter (slope) is constructed during the solution. On the second and all subsequent steps, we proceed as follows (Figure3)

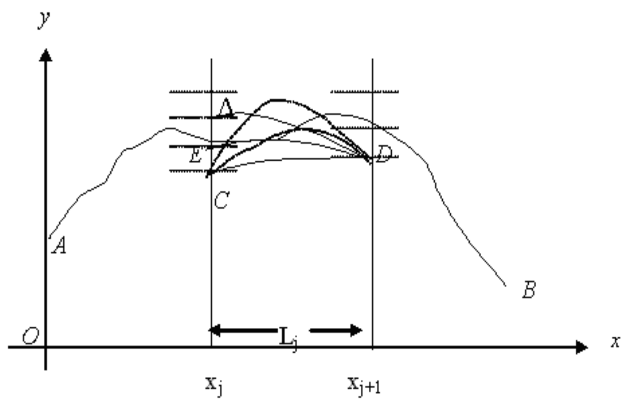


Figure 3. The variants of the next parabola

First Step:

We take the point on last considered vertical (on the second step this is 1 vertical), which has the lowest ordinate (point C). Than we construct the allowable parabola to the lowest point of the next vertical (point D). We leave only allowable options for further analysis and for each of them we calculate the total value of the criterion from the beginning of the curve (point A) to the current point D.

Second Step

We take the next point on the left vertical (point E) and we construct the parabola from this point to point D using one of the initial slopes. But in this case we use the following rule of rejection: if the slopes at the end of the parabolas that converge at the current point (D) are not significantly different ($| i_j''{}_{+1} - i'_{j+1} | < \varepsilon$), then we leave only option with the lower criterion value.

The value of ε plays the role of the discrete for slopes. It should be chosen so as the maximum deviation which arise due to it on the next element does not exceed the grid discrete Δ. The concern is that in the selection of the next parabola we want to know the influence of the small variation of the initial slope, which is used to calculate the parameters. a, b, c. Change i_j causes a change in these parameters, and as a result we get another parabola, but with the same start and end points. The difference between the ordinates of two such parabolas reaches an extreme value in their center. Let us find the maximum deviation within the element between the two parabolas, the initial slopes of which differ by ε. Obviously, this deviation does not change if the origin of coordinates will be placed at the starting point of the element. This significantly simplifies the calculation of the parameters.

Thus, $x_A=0$; $y_A=0$; $b= i_j$; $c=0$; $x_C=L_j$; $y_C=y_{j+1}-y_j$ is known and don't change. For unknown parameter **a** we have $y_{j+1}-y_j = \boldsymbol{a}L_j^2 + i_jL_j$. It follows that $\delta aL_j = -\delta i_j$. Here δi_j - change of the initial slope, and δa consequent change in the parameter **a**. In the mid-point of the element difference between the ordinates of two parabolas with initial slopes i_j and $i_j + \delta i_j$ is equal $\delta\boldsymbol{a}L_j^2/4 + \delta i_jL_j/2=\delta i_jL_j/4$. This difference should not exceed the grid discrete Δ, therefore $\delta i_j<4\Delta/L_j$. We take discrete of slope as half $\varepsilon = 2\Delta/ L_{min}$, where L_{min}-minimum length of the element (on axis OX), because the consideration of the elements that are longer $2L_{min}$ has no special meaning. Each of such element can be constructed as a combination of two elements.

Third Step

By turns we consider connection all points of the left vertical with the point D on the right vertical. We reject a part of allowable connecting parabolas using the rule formulated above. Further we store slopes in the end point D and for each of them total value of the criterion, and the corresponding point on the left vertical.

Fourth Step

We go to the next point on the right vertical and perform the same action.

As a result, we have "fan" slopes at each point on the right vertical, which are considered in the next step as the initial. Naturally, at the last step it is only one point on the right vertical. It is endpoint B. For endpoint B we find the best point on the one of previous verticals, so total value of the criterion will be minimum. And for this point we have stored a slope and a point on the one of previous vertical and so on.

In other words, in the current state (the point plus the slope) were a lot of incoming paths, but we need in only the best one (totally from point A). Therefore, it is necessary to remember number of the vertical, and state on it, which is the beginning of the last element on optimal path to the current state. This information allows to restore the entire line using the reverse movement to the point A.If at the point B the final direction is set, it is necessary to take it into account when calculating the allowable variants of the last parabola.

Now we consider the problem with unknown lengths of the elements. In addition to discrete on vertical Δ and on slopes ε, we introduce the discrete on length of the element λ. The presence of the discreteness is characteristic for the route design and it is a complicating factor in the use of nonlinear programming. In this case, on the contrary, discreteness simplifies the problem.

Possible length of the first element is in the range from Lmin to $2L_{min}$ with step λ. This means that the start point A (Figure 2) may be connected with the points on not only one but on several verticals (Figure 4).

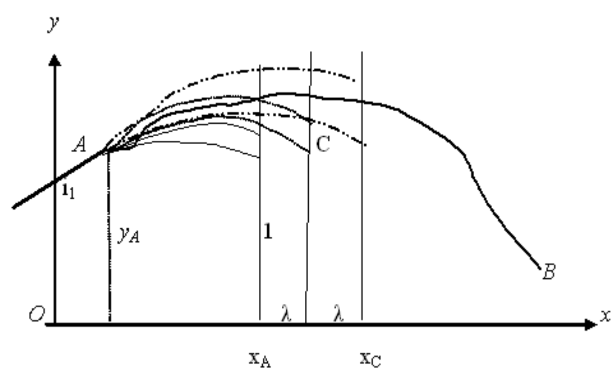


Figure 4. The variants of the first element unknown length

Each of the possible points on these verticals are considered as the beginning of the second element and so on. "System state" is still a point and slope, but the number of verticals ("process steps") is much greater than with the known lengths of the elements.

Moreover, for storing the connection with the previous state we should store this state (the start point and slope at the beginning of the current element) and additionally number of the vertical to which this start point belongs. A step of the process is not a transition to the next element, but to the next vertical, as one and the same point on vertical may be the endpoint of the few paths, which are composed of a different number of elements.

As before the total value of criterion (from point A up to current point) is calculated and stored for each state (point plus slope). We check all restriction and reject the unsuitable variants. The valid paths with a common endpoint, and near values of the final slopes are considered as comparable. In accordance with Bellman's principle of optimality [7,8] from all comparable paths in each state we leave for further consideration only one , for which the total value of criterion is minimal. As before, the problem is two-parameter, but a number of paths is significantly greater than for defined lengths of the elements.

The last vertical is spaced from the endpoint B of approximated curve on L_{min}. After this vertical will be reached we will consider all valid connections of point B with all points on all verticals, which are spaced from point B on $2L_{min}$ and less. Thus we construct the last element CB (Figure 5).

For all valid variants of the last element CB we calculate the value of criterion and add it to the value of criterion which is stored for point C and each incoming slope. All valid path from all states (point C plus incoming slope) to point B are comparable and we find the optimal variant of the last element and minimal value of the total criterion.

The optimal trajectory is restored by moving in reverse. We use number of the point, the slope and number of the previous vertical, which were stored for connection.

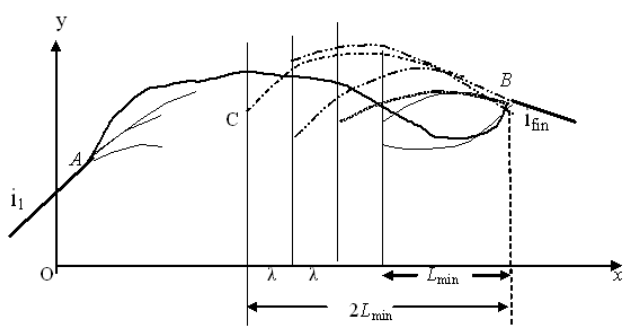


Figure 5. The variants of the last element

Indeed, for start state (point C plus slope) of the selected variant of the last element we know the start state of the last but one element and so on. Thus, as result we will find the number of elements of the optimal approximating curve and all their parameters.

If the neighboring element beyond of the endpoint B and it slope i_{fin} (Figure 5) are given then at the last step when connecting to the point B we take it into account. From all valid variants of the last element we must consider only those whose slope at the endpoint is equal to the slope of this neighboring element i_{fin}. If there is no such elements, then we consider as comparable all elements whose slope at the end point deviates from i_{fin} less than on ε.

4. RESULTS

It is known that the practical application of dynamic programming for a large number of states at each step involves dealing with significant computational complexity which dramatically increases with the number of states [7,8]. This phenomenon is called “curse of dimensionality."

The purpose of experimental calculations was to establish the possibility of practical application of the algorithm for designing the longitudinal profile of new roads on public computers with the search area width 0.5-0.6 m.

Due to the restrictions of type 1 and 2 we cannot perform calculations with a grid with large cells (for example 0.1-0.2m), followed by splitting them into a finer grid depending on the solution obtained. Therefore, we have 25 or more points on each vertical, when width of the search area is 0.5-0.6 m. With the development of computer programs that implement the algorithm for piecewise parabolic approximation, it was possible to substantially reduce the computation time by putting in order the elements of "incoming fan" at each state. It was established experimentally that when the number of discretes on each vertical **k** $\leq$ 30, the number of slopes on each state **m** $\leq$ 60, l_{min}=200 m, λ=10 m and the length of route L$\leq$ 10000m there are not significant computational difficulties. Such tasks were solved on computer with RAM =512 Mb, CPU 2.0 GHz. But computational difficulties for this computer became insuperable if **k=**70, **m**=80, l_{min}=200 m, λ=10 m, L=10000m.

5. DISCUSSION

The proposed algorithm is one element of a multi-stage technology computer-aided design of the longitudinal profile of the roads. This technology is based on a complex optimization methods: nonlinear and dynamic programming. It has clear advantages over the interactive elaboration of design solutions.

The use of complex mathematical models can solve design problems with their relationship and create intelligent CAD. Design of longitudinal profile using optimization algorithms gives a chance for objectively comparison the various plan options and for choosing the most appropriate one. This approach is more promising than the various heuristic algorithms for road routes design.

Computational difficulties with which have to deal, can be overcome in various ways:

1. The use of more powerful computers. But this is not always possible.
2. Decrease in the number of stored parameters for each state. To communicate with the previous state is not necessary to store the slope, enough to remember number of the vertical at the beginning of the element and number of the point on it. But in this case it is necessary to perform additional calculations for restoring the optimal trajectory by reverse moving. For this purpose, we use the formula 2 in reverse order and calculate the slope at the beginning of the current element.

3. On the second stage of projecting involving the road longitudinal profile we don't need in high accuracy of approximation, since we only want to find the number of elements and their approximate position. On the third stage we will get the final decision using non-linear programming. So we may use discrete Δ=0.02 m instead of 0.01m. Accordingly k=30 and m=60 are enough.
4. 4. Using formulae for calculation parabola parameters (1) and end slope (2) we may interrupt the enumeration of the states when some restrictions are not valid. For example if ij+1<imin (see formulae 2) there is no sense in continuing the search with an increase in ij for the current point.

6. CONCLUSION

To apply dynamic programming it is necessary to identify the key concepts: the "step process", the "system state" and give the rule how to construct a set of states at each step, how to reject unpromising paths leading to the current state, and what must be stored to recover the optimal trajectory. It is done in this article with respect to the piecewise parabolic approximation of the plane curve given sequence of points. The described algorithm is used in new CAD system. It is for the design of the road routes. The algorithm is easily generalized to the case when the elements are circles and straight lines. This problem arises in the routing of other linear structures such as pipelines. However the approximation algorithm of discretely defined plane curves with restrictions may find other applications that have nothing to do with designing of routes.

REFERENCES

[1] CARD/1. URL: http://www.card-1.com/en/home/

[2] Topomatic Robur. URL: http:// www.topomatic.ru

[3] Bentley Rail Track. http://www.bentley.com/

[4] C. Parkholup “The system for design of transport highways “Trimble Quantm”, ComputerPress, CAD and Grafics, No 3, 2013

[5] V. I. Struchenkov. “ Optimization Methods in Applied Problems”, Solon-Press, Moscow, 2009.

[6] V.I. Struchenkov “Mathematical Models and Optimization in Line Structure Routing: Survey and Advanced Results”, International Journal Communication, Network and System Sciences. Special Issue : Models and Algorithms for Application, 5, 2012

[7] R. Bellman and S. Drejfus, “Applied Dynamic Programming”, Princeton University Press, Princeton, 1962

[8] E.S. Wentzel “Operations Research: Challenges, principles, metodologiya.” KnoRus, Moscow, 2010

SOCIETY FOR SCIENCE AND EDUCATION
UNITED KINGDOM
TMLAI TRANSACTIONS ON
MACHINE LEARNING AND ARTIFICIAL INTELLIGENCE
Vol. 1 Isuue 1

Design and Implementation of an Online Brain Computer Interface System

M. Raza Fraz[1], Zulqarnain Zahid [1], Shair Shah[2]

[1]*National University of Sciences and Technology, Islamabad, Pakistan;*

[2] *University of West of England, Bristol, United Kingdom;*

Abstract

Brain Computer Interface is the communication channel between the brain and the computer for recording of electrical activity along the scalp produced by the firing of neurons within the brain. The brain signals which are also known as Electroencephalography (EEG) can be used to direct and control some external activity. This work reports a methodology for acquisition and detection and of EEG signals, and extraction of useful information in order to differentiate the signals related to particular type of movement. A modified Common Spatial Pattern (CSP) algorithm has been used at preprocessing stage. Logarithmic transform along with the information theoretic feature extraction has also been used for feature extraction. KNN, SVM and Artificial Neural Networks are employed for classification. The proposed methodology is tested on publically available data sets and the results are found to be comparable with the published approaches.

Keywords—Brain Computer Interface, Common Spatial Pattern, Electroencephalograph (EEG), K-Nearest Neighbor, Support Vector Machine (SVM), Neural Network.

1. INTRODUCTION

BRAIN COMPUTER INTERFACE (BCI) is a communication channel between the brain and the computer which is used to direct and control several types of external activities like controlling some machinery without involving any kind of physical movements. BCI is an emerging field now days and it has vast applications in the field of medical sciences. Patients suffering from severe mental impairments which restrict their movements can make use of this machine to make their mobility possible and easy without dependence on others. This paper focuses on the detection of Brain electrical activity produced by firing of neurons within the brain i.e. the EEG signals and then the overall prediction of the imagined movement.

EEG signals are recorded by measuring the electrical activity of brain using electrodes placed along the scalp. This electrical activity is basically due to electrical fringing of neurons. So

M. R. Fraz, Z. Zahid, S. Shah; *Design and Implementation of an Online Brain Computer Interface System* in Transactions on Machine Learning and Artificial Intelligence, Volume 1, No 1 (2013), PP 27-41
URL*: http://scholarpublishing.org/index.php/TMLAI/article/view/30/TMLAI-13-1025*
Publication Date*: December 17th 2013*

BCI uses EEG signals and works on the surface information. As this is a non-invasive method, the signals recorded are subjected to surrounding noise. Therefore proper filtering and amplification of these signals is required. The first step is of signal acquisition from the surface so they are highly distorted by noise and ocular artifacts. The raw EEG data is preprocessed to get the required brain signals. It's very important to extract the discriminative features from signal, to classify the activity related tasks. After pre-processing and feature extraction classification is done on the basis of feature vector, which is then used to generate control commands to control any external machinery (e.g. wheel chair).

The algorithm frequently used for preprocessing of the data before applying feature extraction and classification is Common Spatial Patterns (CSP) algorithm. CSP was first proposed in 1991 to detect the abnormalities in the EEG signals. Later in 2000, it started to be used for detecting the event related de-synchronizations (ERD) [1]. The imagination of a limb movement modifies electrical activity of brain. The change in this electrical activity is different for imagination of different movements. For example in case of real or imagined hand movement, event related de-synchronizations can be observed in the electrical activity of brain [2]. In case of one-sided hand movement imagination changes the EEG signals recorded from contra and ipsilateral central areas. The patterns of electrical activity so obtained will be different for those obtained in case of left hand movement or any other limb movement. This difference in the electrical activity is not visible if we just look at the signal by naked eye. But with the help of certain algorithm, the two movements can be separated. This forms the basis of CSP algorithm [3, 4].

First we decompose raw EEG data into spatial patterns for the two classes. Then we calculate spatial patterns in such a way so as to maximize the ratio of the variance of data conditioned on one class to the variance of data conditioned on the other class. In this way, spatial filters are designed to extract those components of the EEG data that differs maximally (in terms of variance) between conditions. These spatial patterns are then used to extract features on the basis of log transformation. We proposed a modification in original CSP algorithm that is spatial filters are extracted on the basis of mutual information between the classes. To make selection of the subset of the spatial filters optimum in terms of minimum classification error we are proposing a theoretic framework that is information theoretic feature extraction. In this way not only two classes but multiple classes can be discriminated form each other and classified. We also proposed optimization of certain parameter along with the choice for the optimum classifier which leads to improved accuracies which are further illustrated in detail in this paper.

The paper is organized as follows. In Section II, datasets used and their processing is explained. Extractions of spatial features are explained in Section III. The next section describes the methods used for their classification and proposes the most suitable classifier with CSP. Parameter optimization for the improvement of classification accuracies and results are presented in Section V. Finally, Section VI is dedicated to the presentation of the conclusions of this work.

2. DATA ACQUISITION AND SIGAL PROCESSING

2.1 The Datasets

1.1.1 BCI competition IV

First dataset used (dataset#1 BCI competition IV)[5] was provided by Berlin institute of Technology. These datasets were recorded from healthy subjects in whom motor Imagery was performed without Feedback. For each subject two classes of motor imagery were selected from the three classes‟ left hand, right hand, and foot.

1.1.2 Dataset IVa BCI Competition III

Second dataset used (dataset IVa BCI Competition III) [6] was provided by Berlin BCI group: Fraunhofer First, Intelligent Data Analysis Group (Klaus Robert Muller, Benjamin Blankertz) and campus Benjamin Franklin of the Charite University Medicine Berlin .The datasets are for two class motor imagery (right hand and foot) taken from 5 subjects. Training samples provided are less as compared to the testing samples. Data was recorded from 118 EEG channels at 1000Hz sampling rate .We used the down sampled to 100Hz version of the dataset.

1.1.3 Dataset 2a BCI competition IV

Dataset 2a, BCI competition IV [12], which is a four class dataset, was provided by Institute for Knowledge Discovery, Graz University of Technology, Austria and Institute for Human-Computer Interfaces, Graz University of Technology, Austria. This includes; Class-I (Right hand movement), Class-II (Left hand movement), Class-III (Both feet movement), and Class-IV (tongue).

2.2 Organization of Raw EEG data

Before the implementation of CSP [7] raw EEG data is filtered between 8-30Hz in order to remove the artifacts caused due to eye movements also known as EOG artifacts[8]. Filtration greatly improved the accuracies obtained. Data is filtered using the Butterworth filter of order 10. Two types of filtration is done i.e. Low pass filtering and Band-Pass filtering.

For analysis raw EEG data is arranged into E-matrices. Each E-matrix is of size (N x T), where "N" is the number of channels and "*T*" is the number of EEG samples per channel in a specific interval of time "*T*". So, in this way the raw EEG data is first arranged into structures.

2.3 Co-variance Matrix Formation

Normalized spatial co-variance is obtained by implementing the following equation

$$C = \frac{EE'}{trace\ (EE')} \tag{1}$$

Where $E'$is the transpose of the "E matrix" and "trace (EE′)" is the sum of diagonal elements of EE'. This C is calculated for both classes data individually. For example for the first dataset "C_l" is for left hand imagination data and "C_r" is for the data related to the imagination of right hand class. They are averaged over each class. For the first dataset case "C_l" and "C_r" are both of order (59 x 59). The overall composite spatial co-variance is given by

$$C_c = \overline{C_l} + \overline{C_r} \tag{2}$$

Now "Cc" is factorized into eigenvalues and eigenvectors.

$$C_c = U_c \lambda_c U'_c \tag{3}$$

Where "Uc" is the matrix containing Eigenvectors and λ_c is the diagonal matrix containing the eigenvalues. all the eigenvalues are arranged in the descending order.

2.4 Whitening Transform Projection Matrix and Spatial Patterns Calculation

After this we applied the whitening transformation

$$P = \sqrt{\lambda_c^{-1} U'_c} \tag{4}$$

This whitening transform equalizes the variances in the space that is created by U_c. If we calculate P C_c P' it will result in 1. This will show that up till now our method is correct and we have successfully maximized the variance. After this we have to find

$$S_1 = P\bar{C}_l\, P' \tag{5}$$

Similarly

$$S_r = P\bar{C}_r\, P' \tag{6}$$

Another test to check the procedure is if we take $S_l = B\lambda_l$B' and $S_r = B\lambda_r$B' than λ_l + λ_r= I (identity matrix). Now as we know the sum of the corresponding eigenvalues is always 1, so this test indicates that when λ_l is maximum λ_r is minimum. This indicates that eigenvalues for one class will be maximum at a point whereas the other class will have eigenvalues minimum at that same point. So we are successful in maximizing the co-variance between the two classes. Now B was the eigenvectors. This distribution in covariance makes it possible to classify the eigenvectors belonging to the two different classes. Now in order to find the feature vectors we have to find the projection of "P" (whitening transform), onto the first and last eigenvectors in B that are corresponding to largest λ_l and λ_r. The projection matrix is given by

$$W = (B'P') \tag{7}$$

2.5 Joint approximate diagonalization

In CSP for two class, diagonalization of two covariance matrices is done. Now for M number of classes eigenvectors are combined in a *W* matrix such that

$$W^T R_{x|ci} W = Dci \tag{8}$$

Where, "*R*" is the covariance matrix, ci represents to which class it belongs.

This "*W*" matrix is then used for diagonalizing of covariance matrices of multiple classes .Once this transformation is found some columns of this "*W*" matrix are selected as spatial filters. Up till now it remains ambiguous that which columns are selected as spatial patterns, which will give the most optimum result. For this purpose, the eigenvalues are computed of all the covariance matrices, then eigenvectors which corresponds to the largest eigenvalues are selected. In a case that any eigenvector is selected more than one times that is it corresponds to more than one largest eigenvalues than the eigenvector with next eigenvalue is selected.

3. FEATURE EXTRACTION

3.1 Information Theoretic Feature Extraction

To make selection of the subset of the spatial filters optimum in terms of minimum classification error we are proposing a theoretic framework named as "information theoretic feature extraction". [9].

First of all covariance matrices of each class is calculated as.

$$C_i = \frac{E_i E_i'}{\sum_N E_i E_i'} \tag{9}$$

Where i=1,…,M and N is the number of class

These Covariance matrices are then joined in a single matrix R, which is then computed to get transformation "*W*". First of all an update matrix "*W*" is computed using A Fast algorithm for Joint diagonalization with Non-orthogonal Transformations generally known as "FFDIAG**"** . For the computation of this update matrix following equations are used.

$$Z_{ij} = \sum_k C_i^k \, C_j^k \tag{10}$$

$$Y_{ij} = \sum_k D_j \frac{E_{ij}^k + E_{ji}^k}{2} \tag{11}$$

$$W_{ij} = \frac{Z_{ij}Y_{ji} - Z_{ii}Y_{ij}}{Z_{jj}Z_{ii} - Z_{jj}^2} \tag{12}$$

$$W_{ji} = \frac{Z_{ij}Y_{ij} - Z_{ji}Y_{ji}}{Z_{jj}Z_{ii} - Z_{ij}^2} \tag{13}$$

Then normalization on this update matrix is performed, for this first scale "*W*" by power of two so that its norm is less than one. The orthogonallity of update matrix is ensured by exponentially update of a matrix.

$$V_{n+1} = e^{W_n} V_n \tag{14}$$

Where "W_n" is a skew matrix, i-e $W = -W^T$. Once we get this update matrix, there comes the selection of spatial filters for optimum results.

Then Gaussian mutual information is computed.

$$I_g = \log(w' R_x w) \tag{15}$$

Then estimation of negentropy is performed and subtracted from mutual information, and then "*N*" number of columns of transformation matrix with highest mutual information is selected as most optimum spatial filter to get the final transformation matrix. This transformation matrix is then used to calculate features.

3.2 Logarithmic transform

The features are calculated by decomposition of trails of "*E*" and can be given as

$$Z = WE \tag{16}$$

Where, "*W*" was the projection matrix.

Now for each class EEG sample matrix we are going to select only small number of signals say 'm' that are most important for discrimination between the two classes. As mentioned earlier the discrimination is achieved on the basis of maximized covariance. We are going to select "*Zn*" signals only that will play the most important role in maximizing the covariance, where n=1…….2m. These will be associated to largest eigenvalues (λ_l,λ_r). We will be taking the m first and m last rows of Z so making the total dimension 2m. The feature vectors can be calculated by the following equation.

$$f_p = \log\left(\frac{var\,(Zn)}{\sum_{i=1}^{2m} var(Z_i)}\right) \tag{17}$$

4. . CLASSIFICATION

Three different types of classifiers were used to evaluate the classification accuracies for two, three and four class data. The projection matrix “$W_{N\times D}$"calculated from the training data is used to plot the features for the testing data. Based on the graphs obtained for features the KNN gives the best classification accuracies as compared to the SVM and Neural Network Classifier [10]. Results of different classifiers on dataset IVa of BCI competition III are shows in the *Figure 2*

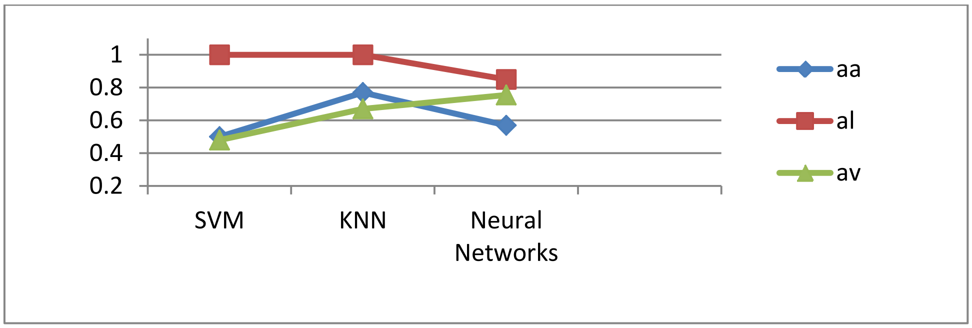


Figure 2: Classification Accuracies using Three Different Classifiers

Hence the KNN classifier was used for two, three and four class classifications and the maximum number of neighbors was selected as five. Parameter optimization is also done to further improve the classification accuracies. All these aspects are further discussed in the subsequent section.

5. RESULTS

Three different datasets taken from the BCI competitions are used for classification of movement related features.

5.1 Two Class Analysis

Dataset IVa, provided by BCI competition III is used for the evaluation of two classes. This dataset contains EEG data from five healthy subjects. The data was recorded without feedback. Visual cue instructing a specific movement was presented for a period of three and a half seconds. After this the movement imagination signal was recorded for three seconds. Data was recorded for the imagination of three motor actions, but only two were released on internet. These two were the imagination of right hand movement and the foot movement. The data was recorded from 118 EEG channels [11]. Total two hundred and eighty samples were recorded and they were divided into training and the testing data.

Table 1: Division of dataset in different subjects

Subjects	aa	al	av	aw	ay
Number of training samples	28	56	84	224	168
Number of test samples	252	224	196	56	112

As mentioned earlier the best results were obtained when the KNN-Classifier was used along with features extracted using log transformation from the selected spatial patterns with maximum mutual information. The extracted features for the two class analysis are shown in Figure 3, where *circles* represents training class 1, '+' represents training class2, *triangles* represents testing class 2 and '*' represents testing class 1.

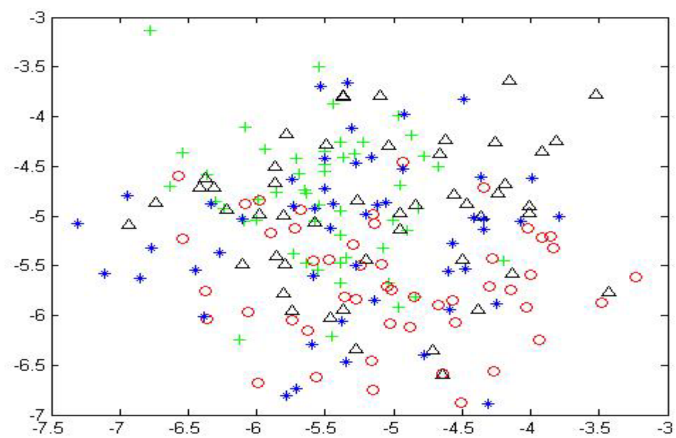


Figure 3: Feature separation

Proposed method is applied instead of original CSP algorithm which results in improvement of accuracies. This improvement in feature separation is illustrated in Figure 4. The dignity of proposed method allows us to use KNN classifier, which is simplest to implement as compare to other two classifiers.

5.2 Parameter Optimization for Classification Accuracies Improvement

For the analysis all the data was divided into E-matrices such that $E_{N\times T1} \in R_{N\times T}$ where "*T1*" is a subset of "*T*". In this way all the EEG data taken from the dataset was divided into equal matrices. Moreover, it was found that if the starting points of the window (Km) then the overlaps in time between the windows and window size (total sample points in window) was varied, and its effect on results is illustrated in Figure 4.

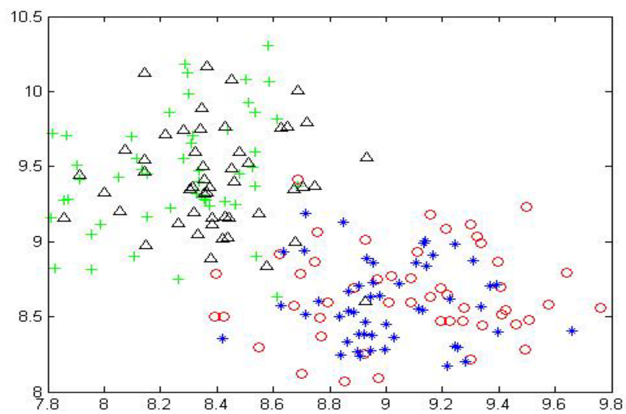


Figure 4: Feature separation

By sliding the window for a trail we select only that portion of the overall imagination trail, which gives the best classification for a given class. Furthermore, the classification accuracies are also affected by varying the dimension N of the feature space which actually corresponds to the number of columns of transformation matrix with highest mutual information. The factor "k" number of nearest neighbor used in classification is also optimized and best classification is obtain when k =2, as illustrated in Figure 5.

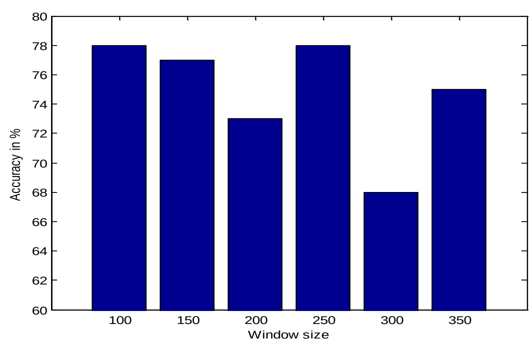


Figure 5: Effect of varying window size on accuracy.

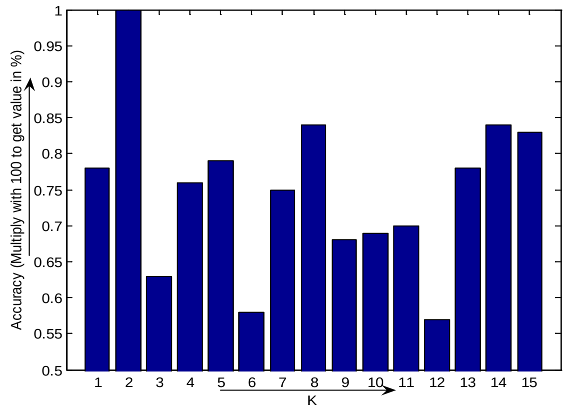


Figure 6: Effect of varying k (number of nearest neighbors) on accuracy

Table 2 shows the best accuracies obtained by varying window size of different subjects.

Table 2 : Best window size accuracies for different subjects

Subject	Window Size	N	Number of Test Samples	Accuracy
aa	100	2	112	77.7%
al	200	2	56	100%
av	250	2	196	75.5
aw	150	2	224	74%
ay	200	2	252	88.9%

Table 3 : Effect of Varying N for subject aa. Best results are obtained for N=2.

Number of training samples	Number of testing Samples	Window Size	Dimension N of feature space	Accuracy
140	140	350	2	92.85%
168	112	350	2	90.476%
168	112	350	6	82.14%
168	112	350	4	80.95%
168	112	350	8	79.76%
140	140	350	118	79.76%

During the analysis it is also observed that if clustering of the data is good then the values of "*k*" (selected nearest neighbors) did not much affect the classification accuracies. For the subject 'aa' the value of "*k*" was varied from 3 to 21 and the accuracies were same keeping all the other parameters unchanged. Analysis was also performed to check whether the classification accuracy depends on subjects or this algorithm is subject independent. Results obtained for two class analyses are summarized in the Table 4, which illustrates that the algorithm is subject dependent.

Table 4 : Subject independent case

Training Subject	Testing Subject	Samples		Accuracy
		Training	*Testing*	
aa,al,av,aw	ay	532	252	65.6%
aa,al,av, ay	aw	504	224	52.2%
aa,al,aw,ay	av	476	196	59.6%
aa,av,aw,ay	al	336	56	50%
al,av,aw,ay	aa	329	112	44.6%
aa,al,av,aw,ay	aa,al, av,aw,ay	560	840	70%

5.3 Multiclass Analysis.

Similar analysis was done for the increased number of classes. KNN-Classifier was the best classifier and again parameter optimization improved the accuracies. Three classes of dataset "*2a BCI competition IV*" are selected for evaluation and parameter optimization. The optimized value of window size was taken as 500. Similarly the value of "*N*" (number of spatial patterns) was selected to be two and four. The value of K for nearest neighbor classifier was five. The results of three class analysis using the mentioned parameters are summarized in Table 5.

Table 5 : Results for three class analysis with equal training and testing samples

Subject	Correctly Detected Classes	Total Training Samples	Total Testing samples	Accuracy
A01T	90	108	108	83%
A03T	95	108	108	87.96%
A04T	65	108	108	60.18%
A07T	88	108	108	81.48%
A08T	95	108	108	87.96%

The nobility of this work is that analysis with greater amount of testing samples than the training samples can be done, which is illustrated in Table 6.

Table 6 : Results for three class using unequal training and testing samples

Subject	Correctly Detected Classes	Total Training Samples	Total Testing Samples	Accuracy
A01T	107	90	126	85.50%
A03T	102	105	111	91.89%
A08T	96	96	120	80.46%

It is observed that different accuracies are obtained for different subjects. Hence it signifies that this algorithm is subject dependent and accuracies greatly depend on the subject's concentration. It is also observed that in this dataset when only three movements imagination (Right hand, left hand, both feet) were considered results were good but including the fourth class i.e. the imagination of tongue movement greatly degraded the accuracies. This can be observed in the Table 7.

Table 7 : Accuracies for four classes

Subject	Training Samples	Testing Samples	Correctly Classified	Accuracy
A01T	144	144	121	84.02%
A03T	144	144	99	68.75%
A04T	144	144	72	50%
A08T	144	144	112	77.77%

5.4 Effective Electrode Reduction.

Using Multiclass CSP algorithm along with information theoretic feature extraction as well as log transformation and KNN classifier resulted in effective reduction of electrodes used to get the brain signals. It is observed that if we optimize the values for the parameters: features space dimension (N), number of nearest neighbors and sliding window, the number of electrodes can be reduced. The offline datasets used for two class analysis used 59 and 118 electrodes (channels) to record the brain signals. In feature space optimization we reduced the dimension space and selected the optimum value of N that gave us the best results (effect of varying N is demonstrated in

Table 3). Actually here we were selecting the minimum number of electrodes required to give best accuracies. It is observed that if we implement the whole procedure on data collected from lesser number of electrodes the results are comparable to those obtained from 118 or 59 numbers of electrodes. Accuracies can be further improved by selecting optimum value of "k" (number of nearest neighbors), greater number of training samples as compared to testing samples and by utilizing the concept of window sliding. Table 8 shows the results when only three electrodes placed at C3, C4 and reference CZ were used to predict the right hand and left hand movement.

Table 8: Classification accuracies using only 3 electrodes

Training Samples	Testing Samples	Correctly detected	Confused	Accuracy
50	6	4	2	66.66%
120	50	33	17	66%
120	6	5	1	83%
120	6(all left)	6	0	100%
120	6(all right)	6	0	100%

This is a subject dependent case and training along with parameter optimization is done every time there is a new subject. So here we get improved accuracies with lesser number of electrodes making it cost effective, but with a disadvantage of greater training requirements. It is suggested that if two movements can be predicted using 3 electrodes only, then 4 movements can be predicted using at least 6 electrodes. Accuracies can be improved by recursive parameter optimization.

6. CONCLUSION

Proper filtration of the EEG data gives significant improvement in the results. The filtration is done in between 8-30Hz, to filter out any kind of artifacts. We are selecting this bandwidth as we are interested in only the alpha α and the beta β region. In this way we train our algorithm to detect the event related de-synchronization, i.e. the change from Mu-Rhythm (α) to active frequency region when the mind is alert/working (β), or in simple words imagining any movement. Filtration is also improvement as it helps the algorithm to get trained on the movement's imagination rather than any artifacts caused by physical movements specifically eye movement. In this way correctly trained algorithm gives good accuracies for testing.

If information theoretic feature extraction is used for the selection of spatial patterns with maximum mutual information along with proper filter implementation and log transformation for features, it gives better separation in feature plotted on a feature space for two classes as compared to conventional CSP algorithms. The features separation can also be improved if we optimize the value for dimensions of the feature space. By dimension N, we actually select only N rows of the spatial patterns that are extracted from a sample of EEG data on the basis of maximum mutual information. It was found that instead of going towards higher dimensions lower values of N also give comparable results with decreased complexity in computations as well. Decreased values of N also assist easier classification algorithm selections. This helps to overall reduce the complexity of design. If N is varied in turn we are selecting lesser number of channels. So it also gives decreased hardware complexity. It was found that rather than using higher number of channels, if only two channels C3 and C4 along with Cz as reference is used it also gives the comparable accuracies. Thus selection of lower value of N not only improves the features representation but also provides an overall simplicity to the design. In this way KNN-classifier also becomes the most effective classifier with CSP in terms of simplicity and accuracies.

Concept of window sliding helps us to select the most optimum window form the training data which gives the best predication of the class label. This window selection is subject dependent as it is selecting a window in which a particular subject EEG gives best results. So window sliding gives improved accuracies but with a constraint of being subject dependent. However if the parameters are first optimized before testing and are then keep fixed for all the subjects then this can help us to choose most appropriate window that can be applied for every EEG sample and get good results.

In summary comparable results can be obtained by a simpler implementation if we first filter the EEG samples, then apply information theoretic feature extraction with lower dimension N instead of conventional CSP and then apply KNN-classifier along with parameters optimization.

REFERENCES

[1] G. Pfurtscheller, “Graphical display and statistical evaluation of event related desynchronization (ERD),” *Electroenc. Clin. Neurophys.*, vol. 43, pp. 757–760, 1977.

[2] S. C. Gandevia and J. C. Rothwell, “Knowledge of motor commands and the recruitment of human motor neurons,” Brain, vol. 110, no. 5, pp. 1117–1130, 1987.

[3] G. Pfurtscheller, C. Neuper, D. Flotzinger, and M. Pregenzer, “EEG based discrimination between imagination of right and left hand movement,” Electroenc. Clin. Neurophys., vol. 103, no. 5, pp. 1–10, 1997.

[4] H. Ramoser, J. Mueller-Gerking, and G. Pfurtscheller, Optimal spatial filtering of single trial EEG during imagined hand movement, IEEE Trans. Rehabil. Eng., vol. 8, no. 4, pp. 441–446, Dec. 2000.

[5] Cedric Gouy-Pailler, Jeremie Mattout, BCI Competition IV, Dataset 1: Motor Imagery, uncued classifier Application, August 2008, France.

[6] Benjamin Blankertz, Klaus-Robert Muller, The BCI Competition III: Validating alternative Approaches to Actual BCI Problems, Neural and Rehabilitation Engineering, 2006

[7] Hyohyeng Kang, Yunjun Nam, Seungjin Choi, Composite Common Spatial Pattern for subject to subject transfer, IEEE Signal Processing letters, Vol. 16, No-8, pages. 683-686, August 2009

[8] G. Pfurtscheller, C. Neuper, D. Flotzinger, and M. Pregenzer, "EEG based discrimination between imagination of right and left hand movement," Electroenc. Clin. Neurophys., vol. 103, no. 5, pp. 1–10, 1997.

[9] Moritz Grosse-Wentrup, Martin Buss, Multiclass Common Spatial Patterns and Information Theoretic Feature Extraction, IEEE transactions on Biomedical Engineering, Vol. 55, page. 1991-1999, August 2008

[10] Artificial Neural networks and digital signal processing, danikos, 2007-2010: http://www.learnartificialneuralnetworks.com/

[11] C.Brunner, R. Leeb, G.R. Muller-Putz, A. Schlogl, and G. Pfurtscheller, BCI competition 2008, Graz dataset A, Institute of Knowledge Discovery, Graz University of Technology, Austria, and Institute for Human-Computer Interfaces, Graz University of Technology, Austria.

[12] Benajmin Blankertz, Klaus Robert Muller, Berlin BCI Group, Datasets-1V(a) BCI IV Description, (motor Imagery, uncued classifier Application), 2008