International Journal of Application or Innovation in Engineering & Management (IJAIEM)
Web Site: www.ijaiem.org Email: editor@ijaiem.org
Volume 3, Issue 7, July 2014 **ISSN 2319 - 4847**

Studies on the performance characteristics of a typical Collision-free Transportation System

Soumen Banerjee[1], Sankhadeep Chatterjee2, Raktim Chakraborty[3]
Hooghly Engineering & Technology College
Vivekananda Road, Pipulpati, Hooghly-712103, West Bengal, India

ABSTRACT

The problem of collision among independent transportation units in a typical transportation network with the help of an efficient collision avoidance algorithm is proposed. The proposed algorithm takes care of a collision-free transportation system in such a way that no two transportation unit will ever collide at any time at any node. It uses a smart prediction mechanism to prevent any possible scheduling that may terminate indefinitely thereby ensuring and preventing any collision what so ever before it can actually materialized. The main objective is fulfilled by integrating the intelligence into source node. In support of its ingenuity, typical experimental results are provided which depicts satisfactory performance in achieving collision-free movement of different independent transportation units in a typical transportation system.

Keyword: Collision Avoidance, Dynamic Reference Set, Signal Set, Priority scheduling, Transportation Network

1.INTRODUCTION

Transportation Engineering, a sub-discipline of Civil Engineering and industrial Engineering, is an application of technology and scientific principles aiming to plan, operate, design and manage different facilities for different transportation modes in order to achieve safe, efficient, flexible, rapid and economical movement of different transportation units in a transportation network. The transportation systems are the building blocks of modern society, are thriving the country's economy and improving quality of life. It finds an extensive application mainly in air, highway, railway, and waterway transportations. It is a matter of utmost concern to achieve safety in all of these different transportation networks to provide a flawless service in respective systems. Collision avoidance has thus become inevitably an imperative part of modern transportation systems in connection to achieving maximum safety in both traditional and intelligent transport systems. It is a rapidly growing technology finding immense application in automobile safety system in reducing the severity of accidents being popularly known as pre-crash system [1], in railway safety and signalling for all railroads [2], in large vessel at sea that are not within range of shore-based system by using AIS, in commercial airlines to avoid mid-air collisions in networking with CSMA and several other fields. In recent times worldwide research is focused on IEEE 802.15.4 / Zigbee protocol was introduced with an objective to solve collision problem in IEEE 802.15.4 LR-WPANs. A novel channel scheduling scheme, an adaptive method with fast recovery [3]. The proposed scheme is aimed at minimizing the possibilities of beacon collision by efficiently managing the multiple available channels in a hybrid manner combining proactive and reactive methods. Surveys on the co-existence between IEEE 802.11 (2.4 GHz Wireless Network) and IEEE 802.15.4-based wireless network are being carried out to underline various issues in the field for further research related to collision avoidance between them [4]-[5]. Studies have also been carried out on different CSMA/CA IEEE 802.11 based implementations. Analysis was done on the access protocol performance in terms of available throughput, access delay and packet dropping. A comparative study on the performance of CSMA/CA vs. CSMA/CD used in wired LANs that are based on IEEE 802.3 standards has also been reported [6]. An efficient method of avoiding train collisions in railway transportation system using vibrating sensors and Zigbee technology is available in literature [7]. Data using wireless communication technology or signals over a part of the entire communication network were transferred. The wireless implementation of sensor network ensures safety in terms of saving lives and property and the collision avoidance system also implemented at various nodes viz., server side node, train side node, track side node and station side node. Also feasibility studies on two methods of crack detection in rail networks and avoidance of collision between the rails have been reported [8]. The collision avoidance algorithm related to state estimation introduces the concept of force field and warning function in the roundabout [9]. Vehicles, by selecting the safety operation mode, avoid the conflict area and pass the merging points by following a specific collision avoidance algorithm. A supervisor based on a hybrid algorithm helping in the synthesis of a least restrictive controller is reported in literature [10] for collision avoidance in case of multiple vehicle at an intersection. Research is also focused on intelligent collision avoidance and adaptive cruise control. Recent developments and research trends in collision avoidance warning system and automation of vehicle longitudinal/ lateral control tasks has been reported [11]. The emphasis is focussed on the initiation for automation in different levels of transportation system with specific emphasise on vehicle level automation. Another interesting application domain of collision avoidance is commercial airlines which witness a series of mid-air collision. Collision avoidance systems for both manned and unmanned aircraft must reliably prevent collision with minimising alerts. The state-of-the-art in Collision avoidance related with air traffic involves the introduction of TCAS (Traffic Collision Avoidance System) and its further modification to provide utmost safety without any sort of interference with normal, safety operation [12]. The robustness of collision avoidance optimization to modelling errors has also been reported [13]. Moreover in cases where only some of the problem dimensions are controllable, research is being carried out in reducing the complexity of computing the optimal strategy [14]. Such an approach finds application in aircraft collision avoidance when the system recommends manoeuvres to an imperfect pilot. Also a new approach for designing safer collision avoidance system, a modification over existing TCAS (Traffic Alert Collision Avoidance System) logic, has been proposed [15]. The approach involves leveraging recent advances in computation to automatically derive optimized collision avoidance logic directly from encounter models and performance metrics with an outline on anticipated impact on development, safety and operation.The Automatic Identification System (AIS), an almost replica of collision avoidance, is basically an automatic tracking system used on ships and by VTS (Vessel Traffic Session) for identifying and tracking of vessels by electronically exchanging data with other nearby ships, AIS base stations and satellites. The AIS system presents collision among large vessels at sea that are not within range of shore based systems. The proposal for the establishment of a regional AIS application specific message register has been reported in literature [16].Thus considering the wide popularity and immense, diversified application of collision avoidance in transportation engineering the authors were inspired to propose a new efficient graph based collision avoidance algorithm using dynamic Reference Set method in resolving the collision among independent network transportation units. The proposed algorithm aims at preventing collision before it happens by centralising the intelligence into source node. The implementation of the proposed algorithm in near future will surely strengthen transportation engineering and build a modern society free from any type of collision at any node at any instance between any two transportation units.

2.DESIGN METHODOLOGY OF COLLISION-FREE TRANSPORTATION SYSTEM

2.1. SIGNAL SET AND DYNAMIC REFERENCE SET IMPLEMENTATION:- A Signal set ($S_{Destination}$) is defined here as a set of nodes of a network such that any node n_i of $S_{Destination}$ may be needed exclusively by any transportation unit E_j at any time T_k to travel from a given source node to a given destination node. A Dynamic Reference Set (R_k), on the other hand, is defined here as set of pair of a node n_i and an transportation unit E_j ((n_i, E_j)) of a Network (N) where each node $n_i \in S_{Destination}$ such that, for any two pair (n_α, E_β), (n_γ, E_δ), $n_\alpha \neq n_\gamma$ and $E_\beta \neq E_\delta$ except Source and Destination at any time instance T_k for any transportation unit E_j, Source, Destination. The authors have designed a new algorithm to eradicate the problem of collision avoidance among the transportation units for a typical collision-free transportation system. The algorithm takes into consideration the smart Set (Signal Set) of nodes ($S_{Destination}$) which is a subset of the set of all nodes (N) of the network and before selecting next node for each transportation unit E_j at any time T_k the algorithm refers the Signal Set which in turn provides knowledge to the algorithm regarding the status of the node. It also utilizes Dynamic Reference Set (R_k), typically needed for knowing the status of nodes belonging to Signal Set, thereby assuring the prevention of faulty allotment of any node n_i of Signal Set ($S_{Destination}$) to any transportation unit E_j for any given Source and Destination. The authors have designed an algorithm for generating the Signal Set for any given destination of any typical network as depicted through Algorithm 1.

Algorithm 1: Signal_Generator (Destination)

//G[1:n][1:n] is the graph depicting the Network (N). Signal Set[1:n] holds the status of Signal Set ($S_{Destination}$). //visited[1:n] is an array which holds the visiting status of the node and all elements are initialized to false, //visited[i] set to true if algorithm has already visited node i.

```
push (Destination)
visited[Destination]←1
do until Stack is empty
    x←pop()
    Signal Set[x]←1
    Repeat until i = | N |
        Check if (G[i][x] ≠ 0 and visited[i] = 0) then
            push(i)
            visited[i] = 1
            end if
        next i
    wend
```

2.2. THE PROPOSED COLLISION AVOIDANCE ALGORITHM

The algorithm designed by the authors considers both the Signal Set and Dynamic Reference Set in scheduling multiple transportation units in a Network in order to avoid collision between multiple transportation units travelling from a given ‘Source’ node to another ‘Destination’ node at any time instance. One fundamental criterion about transportation units is that the transportation units are scheduled in descending order of priorities i.e. for any $\alpha < \beta$, priority of E_α is greater than priority of E_β. The Network is taken as a non-weighted simple graph where each transportation unit takes same amount of time to travel through any edge. The working procedure of the proposed algorithm takes care of the following:

1. The Signal_Set generates the Signal set first for given Source and Destination.
2. The algorithm starts scheduling transportation units for each time instance (T_k) until all the transportation units reach the Destination node.Algorithm 2 depicts the collision avoidance algorithm proposed by the authors for a typical collision-free transportation system. Figure 1 depicts a typical networking system, before being processed by Signal_Generator comprising of a set of nodes N = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10} where nodes '1' and '8' are designated as source and destination respectively. Figure 2 on the other hand represents the same network after being processed with Signal_Generator. The nodes which can be used by any transportation unit E_j at any time T_k are denoted by Green and nodes which must be avoided are kept in red and the executed Signal Set = {1, 2, 3, 4, 5, 6, 8}. Figure 3 is the visualisation of Dynamic Reference Set R_2 at time instance T_2 and is shown by the red coloured bounded region. The transportation units are represented by respective boxes with its nearest node acquired by it thereby creating node-transportation unit pairs. At time instance T_2 transportation unit E_4 possesses node n_2, hence the pair is (n_2, E_4) and so on ultimately leading to a total of 5 such pairs.

Algorithm 2: Collision Avoidance

//G[1:n][1:n] is the graph depicting the Network (N). flag is initialized to 1. num_e is the number of transportation //units and num_v = |N|. Present_Position[i] holds the node acquired by transportation uniti. //Dynamic_Reference_Set[] implements Dynamic Reference Set (R_k). Signal_Set[] implements Signal Set //($S_{Destination}$). t is initialized to 0. It holds the time instance (T_k).

```
until flag = 1 do
  set flag = 0
  repeat for e = 1 to num_e
    if Present_Position[e] ≠ Destination then
      set flag = 1
      repeat for j = 0 to num_v
        if G[Present_Position[e]][j] = 1 and Dynamic_Reference_Set[j] = 0
        and visited[e][j] = 0 and Signal_Set[j] = 1 then
          set Dynamic_Reference_Set[Present_Position[e]] = 0
            set visited[e][j] = 1
            set Dynamic_Reference_Set[j] = e
            Present_Position[e] = j
            break
          end if
        next j
      end if
      set Dynamic_Reference_Set[Destination] = 0
    next e
    set t = t + 1
    if flag = 1 then
      print Present_Position
    end if
wend
```

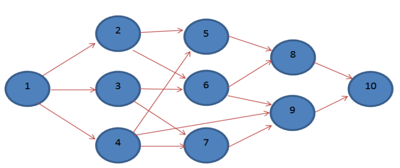

Figure 1: A typical Network (N_1)before Signal_Generator is being applied

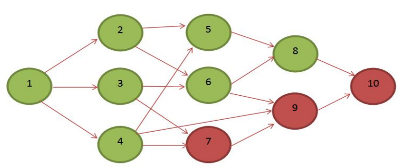

Figure 2: Network of Figure. 1 after Signal_Generator is being applied

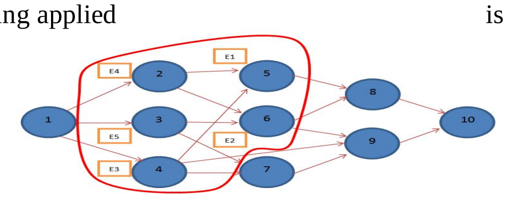

Figure 3: Dynamic Reference Set at Time instance T_2 of Network in Figure. 1 having 8 entities

3. RESULT & DISCUSSION

The priority constraint is kept in mind while proposing the Collision Avoidance algorithm such that for any $\alpha < \beta$, E_α will reach the destination before E_β. Considering scheduling of λ number of transportation units, for all $\alpha < \lambda$, priority of E_λ is less than E_α. Thus E_λ is expected to be the last transportation unit to reach the desired destination. To estimate the maximum time required for one transportation unit E_j ($j \leq \lambda$) to reach the destination the authors have divided the whole network into several layers depending on the Source and Destination node. Here a *Layer* (L_c) is considered as a set of nodes where each node $n_i \in L_c$ and $n_i \in S_{Destination}$ and each n_i is almost at same distance from Source, except Source and Destination provided that for any α and γ there will be no connection between n_α and n_γ. In this context the far and close distance is thought in terms of how much time a transportation unit has taken to travel to one node from another node.

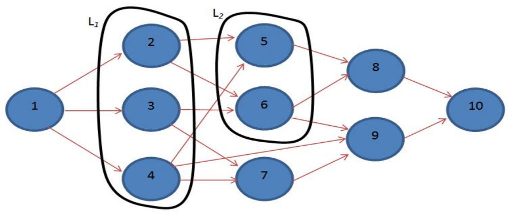

Figure 4: Layering of Network depicted in Figure 1

Figure 4 depicts a typical Layering of Network (Figure. 1) leading to a number of multiple *Layer*s. For instance L_1 = {1, 2, 3} and these nodes can be reached in equal time from Source. A Layer may contain ω number of nodes where $\omega < |S_{Destination}|$ must hold, as the authors are concerned about the intermediate nodes through which transportation units will travel from Source and destination. Thus Source and destination are excluded in any Layer. At least one node of a Layer must be empty such that a transportation unit gets a chance to enter the Layer. For ε no of Layers of a Network, every transportation unit will take ($\varepsilon + 1$) amount of time (Actual Cost) if it doesn't have to wait at any node. But generally if the number of transportation units having greater priorities is at least one then the transportation unit of our interest must have to wait (Redundant Cost) until those higher priority transportation units leave the layer unless next Layer accepts sufficient amount of transportation units at any time instance. Hence total time (Required Time) taken by a transportation unit E_j to travel from Source to Destination is *Required Time = Actual Cost + Redundant Cost*. For a Layer, the Intake Capacity ($I(L_c)$) is the number of incoming paths of that Layer and the Release Capacity ($R(L_c)$) is the number of outgoing paths of the same. There can be multiple transportation units in a Layer at any instance of time T_k where number of transportation units must be less than or equal to the Layer Capacity ($C(L_c)$) which is the cardinality of the set L_c. For a Layer, the source or destination node is reservoir of infinite *Intake Capacity* or *Release Capacity*. A transportation unit can only be released from a Layer L_c iff all the transportation units having higher priorities have already been released from the Layer L_c and have been accepted by the next Layer. Thus if $I(L_c) > R(L_c)$ for any Layer L_c, then $I(L_c) - R(L_c)$ number of transportation units have to wait for at least one time instance in L_c. It is thus evident that if $I(L_c) \leq R(L_c)$ for any Layer L_c, no transportation unit has to wait in L_c. The waiting time of a transportation unit E_j can be calculated taking into consideration the Intake-Release Status (IRS) of the Layers through which it has travelled. An exhaustive study of the IRS of the transportation units is carried out by the authors and one such typical IRS of transportation unit E_5 is reflected in Table 1. *Intake* is the number of transportation units which have been accepted by any Layer L_c at time instance T_k and *Release* denotes the number of transportation units which are leaving any Layer L_c at time instance T_k. The letter 'W' denotes that any transportation unit E_j is waiting in any Layer L_c at time instance T_k while 'T' denotes its travelling status. Thus counting number of 'W's, the total waiting time (Redundant Cost) of that transportation unit can be calculated. Time requirement analysis reflects the maximum Required Time of a transportation unit E_j. The algorithm proposed is tested upon several Networks with different number of transportation units. For instance Figure 5 depicts the scheduling of 9 transportation units in the Network depicted in Figure 1. The Scheduler's table depicts the scheduling of transportation units between Source (1) and destination (8). At any instance of time T_k the Scheduler's table reflects the occupancy of different nodes of different Layers by travelling transportation units. Presence of same colour code at any column of the Scheduler's table indicates that the corresponding transportation unit didn't get the opportunity to leave the Layer due to presence of higher order transportation units or absence of empty nodes in next Layer.

Table1: Intake-Release Status (IRS) of the transportation unit E_5

	Transportation unit Status	Time Instance	Intake	Release
IRS of Source	Waiting (W)	1	-	3
	Travelling (T)	2	-	2
IRS of L_1		1	3	0
	Travelling (T)	2	2	2
	Waiting (W)	3	2	2
	Travelling (T)	4	2	2
IRS of L_2		1	0	0
		2	2	0
		3	2	2
	Travelling (T)	4	2	2
IRS of Destination		1	0	-
		2	0	-
		3	2	-
	Travelling (T)	4	2	-
	Travelling (T)	5	2	-

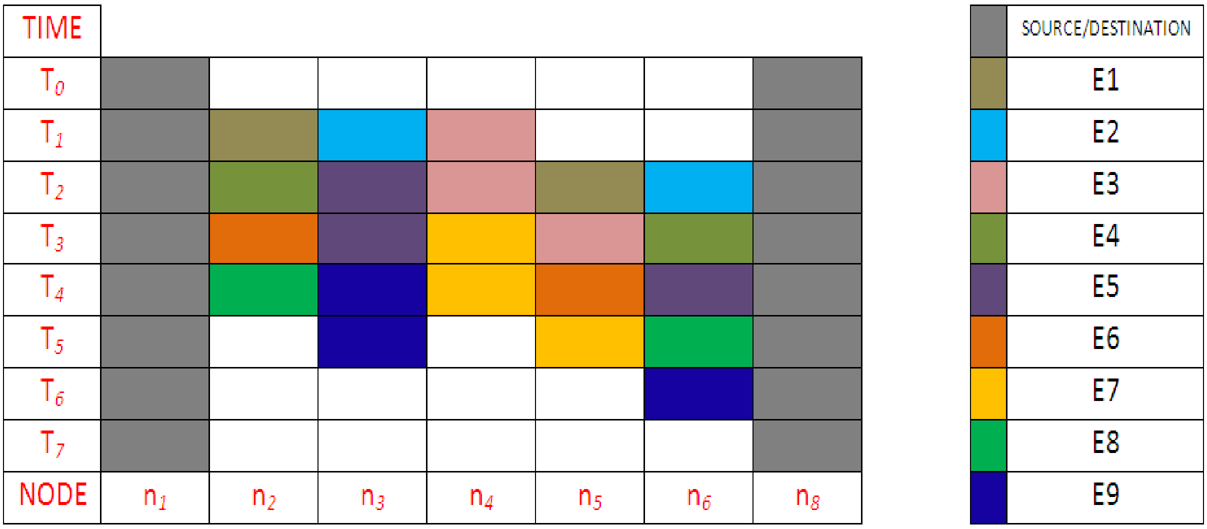

Figure 5: Scheduler's Table: scheduling of 9 transportation units in the Network depicted in Figure 1 with Source '1' and Destination '8'

Figure 6 depicts the allotment status of different nodes of the Network (Figure 1) where Source and Destination have an infinite holding capacity of transportation units. The intermediate nodes, travelled by the transportation units have different *Load*s which signifies the total amount of time for which a node is acquired by any of the transportation units. The figure reveals which node is overloaded in the Network. It is observed that more is the value of 'Busy' time more is the efficiency of that particular node. The analysis hints at optimum utilization of resources in the Network. For example n_5 is 'Busy' for '5' time instances; hence the utilization for it is 62.5% while for n_2 it is 50%. resources in the Network. For example n_5 is 'Busy' for '5' time instances; hence the utilization for it is 62.5% while for n_2 it is 50%.

TIME							
T_0							
T_1							
T_2							
T_3							
T_4							
T_5							
T_6							
T_7							
NODE	n_1	n_2	n_3	n_4	n_5	n_6	n_8
LOAD	∞	4	5	4	4	5	∞
	SOURCE/DESTINATION				FREE		BUSY

Figure 6:- Allotment status of Different nodes of Network depicted in Figure 1 with 9 transportation units.

The graphical representation of scheduling of 9 different transportation units in the Network (Figure 1) is as depicted in Figure 7. The graph reveals a plot of node vs. time instance where the time instance starts from ‘1’ and reflects the status of each transportation unit along with corresponding nodes. It also expresses the waiting status of the transportation units as they travel through the network. For example at time instance ‘2’ E_1 is at node ‘2’, E_2 is at node ‘3’, E_3 is at node ‘4’ rest of the transportation units are still at Source (node ‘1’).

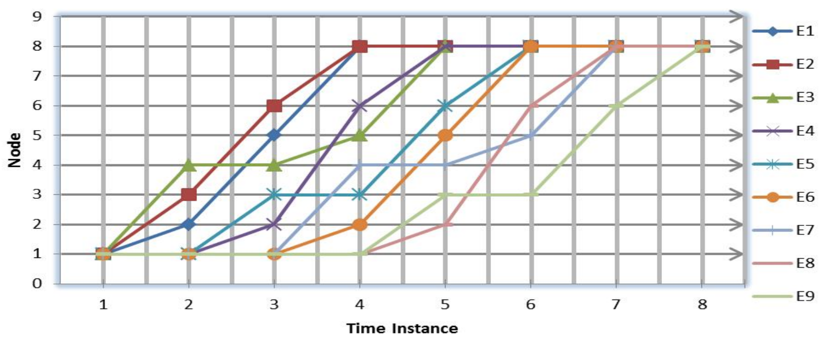

Figure 7:- Graphical representation of movements of transportation units in the Network depicted in Figure 1 having 9 transportation units.

4.CONCLUSION

For a typical transportation system, an absolute new approach has been reflected by the authors in order to avoid any possibility of collision. The collision avoidance algorithm designed aiming at scheduling independent transportation units in preventing collision is a novel approach and has been tested successfully. The performance characteristic of the proposed algorithm gives a very satisfactory result. The proposed Collision avoidance based algorithm maximizes the utilization of resource in a typical transportation system by analysing the time requirement of a transportation unit to travel from source to destination. The proposed methodology is elaborative and shows no collision among the different travelling independent transportation units. The present work is novel of its kind and will be of great help in near future to the worldwide research community in opening a new window in the Transportation Engineering domain.

REFERENCES

[1] http://spectrum.ieee.org/green-tech/advanced-cars/keeping-cars-from-crashing
[2] http://www.argeniarailwaytech.com
[3] Jin-Woo Kim, Yeonwoo Lee, Seong Ro Lee, “Interference Aware MAC scheduling for Collision Avoidance using Energy Detection Scan”, International Journal of Software Engineering and Its Applications, 7(3), pp. 239-248, 2013.
[4] D Yang, Y Xu, M Gidlund, “Wireless Coexistence between IEEE 802.11- and IEEE 802.15.4- based networks, A survey”, International Journal of Distributed Sensor Networks, 2011.
[5] K Shuaib, M Boulmalf, F Sallabi, A Lakas, “Co-existence of Zigbee and WLAN, a performance study”, Wireless Telecommunications Symposium (WTS-2006), pp.1-6, 2006.

[6] Miquel Oliver, Ana Escudero, "Study of different CSMA/CA IEEE 802.11-based implementations", EUNICE 1999 Contribution.
[7] G.Anjali Bissa, S Jayasudha, R Narmatha, B Rajmohan, "Train Collision Avoidance System Using Vibration Sensors And Zigbee Technology", International Journal of Research in Engineering & Advanced Technology, 1(1), pp. 1-7, 2013.
[8] S Ramesh, "Detection of Cracks and Railway Collision Avoidance System', International Journal of Electronic and Electrical Engineering, 4(3), pp. 321-327, 2011.
[9] Lei Wang, Wei Hwang, Xiaomei Liu, Yantao Tian, "Vehicle Collision avoidance algorithm based on state estimation in the roundabout", In the Proceedings of 3rd International Conference on Intelligent Control and Information Processing (ICICIP), pp. 407-412, 2012.
[10] A Colombo, D D Vecchio, "Efficient algorithms for collision avoidance at intersections", In the Proceedings of the 15th ACM International Conference on Hybrid Systems: Computation and Control, pp. 145-154, New York, USA, 2012.
[11] A Vihidi, A Eskandarian, "Research Advances in intelligent collision avoidance and adaptive cruise control", IEEE Transaction on Intelligent Transportation System, 4(3), pp. 143-153, 2003.
[12] Mykel J Kochenderfer, Jessica E Holland, James P Chryssanthacopoulos, "Next-Generation Airborne Collision Avoidance System", Lincoln Laboratory Journal, 19(1), pp. 17-33, 2012.
[13] Mykel J Kochenderfer, James P Chryssanthacopoulos, P. P Radecki, "Robustness of optimized collision avoidance logic to modelling errors", In the Proceedings of IEEE/AIAA Digital Avionics Systems Conference (DASC), pp. 3.B.3-1 – 3.B.3-10, 2010.
[14] Mykel J Kochenderfer, J Chryssanthacopoulos, P James, "Collision avoidance using Partially controlled Markov decision processes", International Conference on Agents and Artificial Intelligence, Rome, Italy 2010.
[15] Mykel J Kochenderfer, J Chryssanthacopoulos, P James, Roland E Weibel, "A New Approach for Designing Safer Collision Avoidance Systems", In the Proceedings of 9th USA/Europe Air Traffic Management Research and Development Seminar (ATM2011), 2011.
[16] Lee Alexander, Kurt Schwehr, "Establishing a Regional AIS Application Specific Message Register", 7th IALA Conference, Cape Town, South Africa, pp. 108-115, 2010.

Soumen Banerjee is Professor and Head of the Department of Electronics & Communication Engineering at Hooghly Engineering & Technology College, West Bengal, India. He obtained B.Sc, B.Tech and M.Tech degrees from University of Calcutta, India. His research interests are microwave & millimeter-wave semiconductor devices & systems, computer network, optimization techniques and artificial intelligence. He has published many papers in International Journals and International/National Conferences. Prof. Banerjee is a member of IET (UK), IETE (New Delhi, India) and IE (India).

Sankhadeep Chatterjee is pursuing B.Tech Course in Computer Science & Engineering from Hooghly Engineering & Technology College, West Bengal, India. His areas of interest are computer network, design and analysis of algorithms, artificial intelligence and optimization techniques. He has published papers related to the field of computer science in International Journals & Conferences.

Raktim Chakraborty is pursuing B.Tech Course in Computer Science & Engineering from Hooghly Engineering & Technology College, West Bengal, India. His areas of interest are database management systems, design and analysis of algorithms, software engineering and computer organization and architecture. He has published papers related to the field of computer science in International Journals Conferences