ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 2, February 2015

Restoring Connectivity of Wireless Networks

S.Pothumani[1], J.Sridhar[2]

Assistant Professor, Dept. of CSE, Bharath University, Chennai, Tamil Nadu, India

ABSTRACT: Wireless sensor and actor networks (WSANs) applications has a set of mobile actor nodes, which are deployment along with the sensors in order to collect sensors' data and perform specific tasks in response to detected events/objects. In most scenarios, actors have to respond collectively, which requires interactor coordination. Therefore, maintaining a connected interactor network is critical to the effectiveness of WSANs. However, WSANs often operate unattended in harsh environments where actors can easily fail or get damaged. An actor failure may lead to partitioning the interactor network and thus hinder the fulfilment of the application requirements. In this paper, we present Actor Replacement (AR) Algorithm, which opts to efficiently restore the connectivity of the interactor network that has been affected by the failure of an actor. Two variants of the algorithm are developed to address 1- and 2-connectivity requirements. The idea is to identify the least set of actors that should be repositioned in order to re-establish a particular level of connectivity. ARA strives to localize the scope of the recovery process and minimize the movement overhead imposed on the involved actors.

I. INTRODUCTION

WIRELESS sensor and actor networks (WSANs) have attracted lots of interest in recent years due to their potential use in numerous applications such as boarder protection, battlefield reconnaissance, space exploration, search and rescue, etc. A typical WSAN consists of a larger set of miniaturized sensor nodes reporting their data to significantly fewer actor (actuator) nodes [1]. Sensors probe their surroundings and report their findings to one or multiple of actors, which process the collected sensor readings and respond to emerging events of interest. An actor's response would depend on its capabilities, which vary based on the application and the expected role the actor plays. For example, an actor can deactivate a landmine, extinguish a fire, and rescue a trapped survivor. It is worth noting that a heterogeneous set of actors may be employed and assigned complementary roles. In most application setups, actors need to coordinate with each other in order to share and process the sensors' data, plan an optimal response and pick the most appropriate subset of actors for executing such a plan. For example, in forest monitoring applications, fire-extinguishing actors need to collaborate with each other in order to effectively control a fire and prevent it from spreading. The selection of actors that need to be engaged can be based on many factors such as the actor's capabilities, actor's proximity to the detected event, and actor's current load. All of these factors would require a frequent update of the actor's state. To enable such interactions, actors need to stay reachable from each other. In other words, a connected interactor network has to be maintained at all times.

In this paper, we study the impact of a node failure on the interactor connectivity in WSANs. Here ARA, an Actor Replacement Algorithm is presented, which opts to efficiently restore the connectivity of an interactor network to its pre-node-failure level. Based on the type of connectivity considered, two algorithms, namely, ARA-1C and ARA- 2C, are developed to address 1 and 2-connectivity requirements, respectively. ARA is a localized scheme that avoids the involvement of every single actor in the network. ARA pursues coordinated multi actor relocation in order to re-establish communication links among impacted actors. The main idea of ARA-1C is to replace the dead actor by a suitable neighbour. The selection of the best candidate (BC) neighbour is based on the node degree and the physical proximity to the dead actor. The relocation procedure is recursively applied to handle any actors that get disconnected due to the movement of one of their neighbours (e.g., the BC that replaced the faulty actor). Similarly, ARA-2C identifies the nodes that are affected, i.e., lost their 2-connectivity property, due to the failed actor. Some of these nodes are then relocated in order to restore 2-connectivity. Although both ARA-1C and ARA-2C pursue node relocation to restore the desired level of connectivity, they fundamentally differ in the scope of the failure analysis and the recovery. The main optimization objective of ARA is to minimize the total distance travelled by the involved actors in order to limit the overhead incurred by the movement. In addition, ARA strives to minimize the messaging costs in order to maintain scalability. The entire recovery process is distributed, enabling the network to self-heal without any external supervision. ARA is validated analytically and through simulation. The simulation results shows that AR is both efficient in achieving minimal total travelled distances and lightweight in terms of required communication resources.

II. TECHNIQUES AND ALGORITHMS USED

A. Re-establishing connectivity by actor replacement

This section investigates effective means for restoring 1- connectivity of WSANs that gets partitioned due to failure of only one particular actor. We opt to pursue a localized, distributed, and lightweight approach in order to suit the nature of WSANs. In this section, we describe the recovery problem and the proposed ARA-1C approach.

a) *Mobile Actor Connectivity Problem*

The impact of actor's failure on the network connectivity can be very limited and can also be very dramatic. When the lost actor is a leaf node, no other actors will be affected. Meanwhile, when the failed actor serves as a cut-vertex node in the network, playing the role of a gateway between two sub networks, a serious damage to the network connectivity will be inflicted. Basically, with the loss of a cut vertex, the network gets partitioned into disjoint sub networks. In order to tolerate the actor failure that causes network partitioning, two methodologies can be identified: 1) precautionary and 2) real-time restoration. The precautionary methodology strives to provision fault tolerance in the network topology both at setup and during normal operation. The idea is to establish a k-connected topology such that every node can reach other nodes over at least k independent paths. Such arrangement will allow the network to seamlessly tolerate the failure of up to k _ 1 actors [4]. As we mentioned earlier, provisioning a high level of connectivity may require the deployment of a large number of actors and may thus be impractical due to their high costs. In addition, it may constrain the mobility of actors and negatively affect application- level functionality. This paper consider the case of 2-connectivity in Section 5. On the other hand, real-time restoration implements a recovery procedure when an actor failure is detected. Here a real-time restoration better suit WSANs is argued since they are asynchronous and reactive in nature and it is difficult to predict the location and the scope of the failure. Therefore, adaptive schemes can best scope the recovery process depending on the effect on the failure on the network connectivity.

B. Detailed ARA-1C Steps

ARA-1C is a distributed algorithm that requires little coordination. As described in the previous section, ARA-1C relies on the neighbours of a failed node for initiating the recovery process. Since these nodes will not be able to communicate after the failure of the cut-vertex node that used to connect them, they cannot coordinate with each other. Instead, they perform the same steps. ARA-1C guarantees that the recovery process converges. The following are the detailed steps.

a) Heartbeats and Neighbour List Maintenance

The only prefailure knowledge that ARA-1C requires for each actor to have is the list of neighbouring actors. Each actor in the network should be aware of its one-hop and two-hop neighbours and should strive to keep this information current. The list of neighbours can be updated each time one of these neighbours changes its position.[1] Under normal conditions, an actor can move to enhance performance, conduct application-specific action, etc. [5]. The two-hop neighbor list is used in identifying the actor that may need to replace a failed node and in optimizing the recovery process. Each actor will also tabulate the node degree, position, and ID of all its neighbours. The two-hop neighbour information table, which we refer to thereafter as TwoHopTable, will be used to identify cut vertices in the network using distributed algorithms like the one proposed in [6].

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 2, February 2015

```
ARA-1C(ID, missing_node)
repeat
  BestCandidateID ← FindBestCandidate(TwoHopTable)
  if ID = BestCandidateID then
    MoveToLocation(ID,loc(missing_node))
    exit
  else
    Remove BestCandidateID from TwoHopTable
    pause( 2 * time to travel for distance r)
  end if
until Msg ('RECOVERED') is received

MoveToLocation(ID, newloc)
if Neighbor(ID) ≠ NULL then
  for each j ∈ Neighbor(ID)
    if IsDependentChild(j)== True then
      Unicast(j, Msg('MOVING', Siblings(ID))
  // All dependent children will then invoke
  // ChildMovOptimizer(j, ID, Siblings(ID))
    end if
  end for
end if
Move(newloc)
Broadcast(Msg('RECOVERED'))
Update TwoHopTable

ChildMovOptimizer(ID, parent, parentSiblings)
for each k ∈ TwoHopTable // meaning ID.TwoHopTable
  if k ∈ parentSiblings then
    exit
  end if
end for
if ID has not previously moved then
    ARA(ID, parent)
end if
```

FIG1. PSEUDOCODE FOR ARA-1C

This type of algorithms generally trade off the need for a network-wide state with the accuracy of identifying cut vertices. It has been shown that the probability of missing a cut vertex is zero while a very high percentage of the picked nodes are really cut vertices [7]. Using two-hop information, it was shown that the accuracy can reach 90 percent [7]. The TwoHopTable will be updated by the individual nodes. Actors will periodically send heartbeat messages to their neighbors to ensure that they are functional and also report changes to the one-hop neighbors. Obviously, for a node B that is a neighbor to A, the one-hop neighbors of A other than B itself are among the two-hop neighbors of B. Formally, we have

1-hop-Neighbors (A) - {B} ⊆ 2-hop-Neighbors (B).

b) Detecting Actor Failure and Initiating the Recovery Process

Missing heartbeat messages can be used to detect the failure of actors. Depending on the actor's position in the network topology, major or no recovery may be needed. As discussed in the prior section, the failure of a node that does not hinder the connectivity of other nodes would not necessitate any adjustments to the network topology. Despite the fact that failure of such a node will not cause any problems to the interactor connectivity, it can create other problems such as forming coverage holes, disrupting the data collection from that particular region, etc. In such cases, depending on

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 2, February 2015

the application-level requirements, these problems need to be handled. We would like to note, however, that handling such problems is out of scope of this paper. We only focus on restoration of interactor connectivity when a cut-vertex node fails. Such a failed cut vertex will trigger the execution of ARA-1C. The entire detection and recovery process is localized. Actors that detect the failure of one of their neighbors will decide on the scope of the recovery. Again, if the failed actor is a cut vertex that causes the network to partition, the restoration process will be initiated on these neighbouring actors. The failed actor is referred to thereafter as Af.

c) Best Candidate Selection

ARA-1C restores the connectivity of a partitioned network by substituting Af with one of its one-hop neighbors. This way, all broken links can be re-established, and potential ambiguity about the signal propagation conditions is avoided. The latter is handled by moving the substitute actor to the exact position of Af . The obvious question is which neighbor should be picked. ARA-1C strives to identify the BC for replacing Af using the following criteria in order:
1. *Least node degree*. The rationale is that the impact of moving a node that has many neighbors will be significant. Thus, ARA-1C favors replacing the failed actor with the neighbor that has the least node degree.
2. *Closest proximity to failed actor*. In order to minimize the movement overhead, the nearest actor to Af will be favoured.
3. *Highest actor ID*. It is possible that among the neighbors of Af, two or more actors have identical node degrees and are equidistant to it. The actor with the greatest ID will be picked to break the tie.

d) Cascaded Node Relocation

The BC actor will prepare itself to move to the location of Af and calculate the expected time it will take to reach the new location. In addition, before moving to the new location, the BC will inform all actors in the set Dependents(BC;Af) about its movement and the time it will take to reach to the new location by sending a "MOVING" message. The BC will then broadcast a "RECOVERED" message upon arriving at the destination.

C. Maintaining 2-connectivity

This section focuses on restoring 2-connectivity after the failure of an actor. The interactor network is assumed to be initially biconnected. The following sections describe the 2- connectivity restoration problem and present the proposed ARA-2C algorithm.

a) Problem Analysis and Approach Overview

As shown in Section 4, when a critical actor fails, some of the nodes may be temporarily isolated until the network connectivity is restored, and thus, the network operation may be disrupted during the recovery. A way to avoid such short-lived disruption is to establish a 2-connected interactor network after deployment. In a 2-connected network, there are at least two node-independent paths among each pair of nodes. This type of connectivity would ensure continual interactor coordination even if an actor fails. Such robust operation is particularly important in WSANs where real-time decisions are made and when the network operates on remote and inaccessible areas such as other planets. Since the network should sustain such robustness throughout its lifetime, 2-connectivity should be restored after an actor fails. The failure of a particular actor may or may not cause the network to lose its 2-connectivity property.

b) Detailed Steps of ARA-2C

ARA-2C has mainly three steps. In the first step, an actor failure is detected, and the nodes that need to get engaged in the recovery process are determined. The second step determines where to move the nodes if deemed necessary. And finally, in the third step, the movement is performed.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 2, February 2015

1. Detecting Failure and Initiating the Recovery Process

Similar to ARA-1C, actors will periodically send heartbeat messages to their neighbours to ensure that they are functional. Missing heartbeat messages repetitively can be used to detect the failure of actors. The neighbours of a failed actor Af will individually decide on the scope of the recovery. Per Theorem 7, the restoration process will be initiated only on every actor Aj ∈ Neighbors (Af) based on the following criteria:

i) Af is a boundary node,

ii) Aj is a boundary node, and

1. iii) the failure of Af introduces a cut vertex in the network. The latter condition can be dropped to limit the required processing at the expense of occasionally performing recovery when not needed. Nodes that satisfy these conditions are referred to as candidates thereafter. ARA-2C restores the 2-connectivity of a network by relocating one of these candidates. *Selecting Which Node to Move and Where to Move It*

ARA-2C employs the results of Theorem 7 to reconnect the neighbours of Af that are boundary nodes (candidates) to each other. To do so, ARA-2C sets selection criteria that uniquely identify the best choice among these candidates to move. Each candidate will refer to its TwoHopTable and find out whether it is the most qualified BC. The main criterion for picking the BC is the following:

i) Lowest node degree. The rationale is that a node that has
the least number of neighbours will yield few cascaded relocations. Obviously, this is not always the case, and it will depend on the number of two-hop neighbours, three-hop neighbours, etc., and how many of them will need to move. We argue that establishing a breadth-first-search tree routed on each candidate would expand the scope of the recovery and would require that each node maintains a network-wide state; something ARA-2C is geared to avoid. The decision on where the BC will reposition depends on the cardinality of the set of candidates. If there are only two candidates, the selected BC will move toward the other candidate until it becomes within its radio range. However, if the cardinality of the set of candidates is three or more, the BC will move to the position of Af. The rationale is that replacing Af is sure to re link all its neighbours and restores the lost connectivity. It is worth to emphasize that we assume that all actors have the same radio range. In case multiple candidates have the least node degree, the following criterion is employed to qualify the best choice[2]. Again, this applies only if there are more than two candidates.

ii) Least distance. In order to minimize the travel overhead, the closest candidate to Af , among those having the least node degree, is favoured for BC. Finally, if there is still a tie, the following criterion is applied

iii*) Highest actor ID.* It is possible, although rare in practice, that there exist two or more candidates with identical node degrees and equidistant to Af. In that case, the candidate with the greatest ID will be picked to break the tie. Note that this also applies for the case in which there are only two candidates, and both have the same node degree.[3] It is important to note that ARA-2C is executed simultaneously on all neighbors of a failed actor without any coordination among them.

FIG2.PSEUDOCODE FOR ARA-2C

c) Node Relocation

The BC actor, A_{BC}, will notify all its neighbors that it is moving and tell them where it intends to reposition[4]. The new position can be determined as explained above. Similar to ARA-1C, when A_{BC} stops, it broadcasts a "RECOVERED" message indicating the completion of the restoration process. The neighbors of A_{BC} will keep waiting for the "RECOVERED" message; if they receive it, they conclude that they are still connected. Otherwise, they will assume that they got disconnected and apply Theorem 7 again as if A_{BC} stopped functioning. This means that a node, Aj Neighbors (A_{BC}) will move only if it is a boundary node. In other words, the recovery process will be applied recursively to trigger the cascaded relocation of affected actors. The only difference here is that each of these boundary nodes will individually move toward A_{BC}.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 2, February 2015

III. PERFORMANCE EVALUATION

A. Performance Evaluation of ARA-1C

In the experiments, we compared ARA-1C and DARA-1CLF to the optimal cascaded motion solution that provides the least travelled distance. The optimal cascaded motion is a centralized approach that requires the knowledge of the state of the entire network. After identifying cut vertices in the generated actor topology, one of them is picked at random to be a faulty node in the network.[5]

Movement performance. In order to assess the effectiveness of ARA-1C and ARA-1CLF in terms of the total travelled distance, we conducted experiments with varying numbers of actors. The results shown that the total distance travelled by actors in ARA-1C and ARA-1CLF is exactly same. The travel distances achieved by ARA-1C get closer to the values obtained by the optimal cascaded motion when the network size grows, confirming the scalability of our approach. This can be attributed to the fact that with increasing the number of actors, the degree of connectivity in the network increases and, thus, fewer movements are required to restore connectivity. It is also interesting to note that local flooding in ARA-1CLF has not brought any advantages as the network is randomly deployed.[6] We have repeated the same experiment with varying transmission ranges while fixing the number of actors at 60. The results shown that the total travel distance grows when increasing the actor radio range. Such results may seem puzzling at the first look since the network connectivity usually improves with greater transmission ranges. However, cut vertices in this case are usually linking blocks that are far apart, and thus, the recovery will involve a travel for longer distances.[7] We have also conducted some experiments with special topologies (i.e., circular connected network) in order to assess the advantage of ARA-1CLF over ARA-1C in terms of total distance traveled. In such cases, ARA-1CLF performed significantly better than ARA-1C, almost matching the performance of the optimal cascading. This is expected as in circular topologies,ARA-1C unnecessarily relocates all the nodes that are not indeed needed for maintaining the connectivity.[8]

Message complexity. In order to compare the messaging overhead, we have recorded the total number of messages sent by all actors in ARA-1C and ARA-1CLF. Table 1 provides the statistics with varying numbers of actors and the actor radio range set to 100m. Such statistics confirm that ARA-1C and ARA-1CLF introduce significantly less interactor communication overhead than the optimal cascaded motion. This is more evident when the actor count is relatively large. The optimal cascading approach requires each actor to flood the network, leading to a message complexity of N^2. While it is conceivable to form a multicast tree in order to reduce the message count, forming this tree is still $O(N^2)$.

TABLE 1
TOTAL NUMBER OF MESSAGES SENT WITH VARYING NUMBERS OF ACTORS

# of Actors	Optimal Cascading	ARA-1C	ARA-1CLF
20	400	87	98
40	1600	168	200
60	3600	249	299
80	6400	305	401
100	10000	406	499

provides the statistics with varying numbers of actors and the actor radio range set to 100m. Such statistics confirm that ARA-1C and ARA-1CLF introduce significantly less interactor communication overhead than the optimal cascaded motion. This is more evident when the actor count is relatively large[9]. The optimal cascading approach requires each actor to flood the network, leading to a message complexity of N^2. While it is conceivable to form a multicast tree in order to reduce the message count, forming this tree is still $O(N^2)$. On the other hand, the message complexity of ARA-1C and ARA-1CLF is linear in the number of actors, as seen in Table 1. Note that since ARA- 1CLF requires local flooding, the number of messages sent in this

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

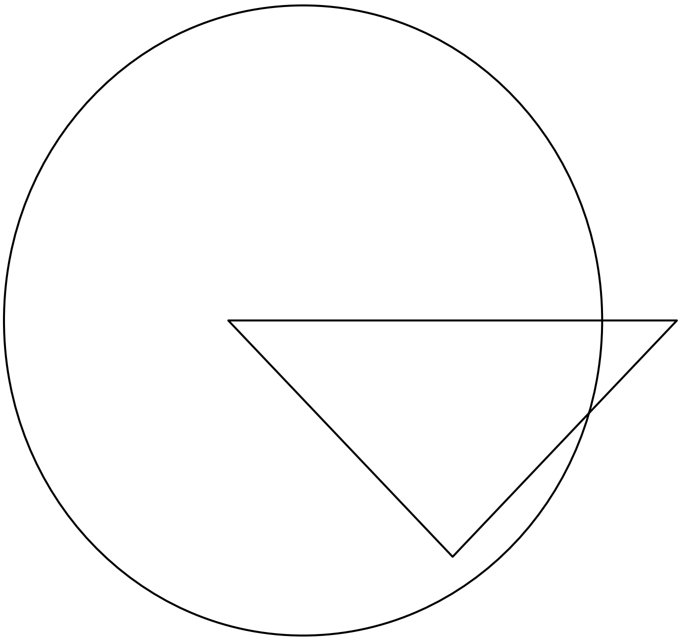
(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 2, February 2015

approach is a little higher than ARA-1C. However, it brings reduction in the total distance movement for certain network topologies, as mentioned above[10].

B. Performance Evaluation of ARA-2C

For these experiments, we created 2-connected interactor networks consisting of varying numbers of actors (20 to 100) using the algorithm in [2].



FIGE3. DISTANCE COVERED BY VARYING ACTOR

We compared the base and simplified versions of ARA-2C to the optimal solution that provides the least movement distance and to the p-Hop algorithm [3], which moves some nodes in order to achieve biconnectivity of an initially 1-connected network. The symbol "p" signifies the number of hops up to which a node is aware of. Since ARA-2C utilizes two-hop information, we compare it to the two-hop version of the algorithm in [3]. Movement performance. We conducted experiments with both varying numbers of actors and varying transmission ranges. The results depicted show that both versions of ARA-2C significantly outperform the p-Hop algorithm in terms of the distance that actors collectively travel to restore the biconnectivity of the network. In many configurations, the base version delivers performance that is at least 300 percent better than the p-Hop algorithm. Even in the simplified version, actors often need a travel distance that is 60 percent-70 percent of that travelled under the p-Hop algorithm. The figures also indicate that the base version of ARA-2C delivers performance that is 25 percent-40 percent better than that of the simplified version. The network designer is sure to weigh such performance advantage against the corresponding increased cost in processing and communication.[11]

IV. CONCLUSION

WSANs are usually deployed in harsh environments where actor failure/damage is possible. Such failures can lead to a situation in which the interactor network connectivity is weakened and the network gets partitioned. In order to recover from these failures, we have presented ARA, a Distributed Actor Recovery Algorithm for WSANs. Two variants of the algorithm, namely, ARA-1C and ARA- C, are developed to address 1 and 2-connectivity requirements. The main goal

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 2, February 2015

of ARA-1C is to restore the network connectivity by exploiting the mobility of actors. The idea is to pick one of the neighbours of a failed node in order to replace it. Since this may still disconnect the children of the moved actor, ARA-1C strives to pick the neighbouring actor that will trigger the least number of cascaded movements.

In this way, the total distance travelled by actors is minimized. ARA-2C strives to restore the 2-connectivity of a 2-connected interactor network. The effect of an actor failure is analyzed and shown to depend on whether the actor is a boundary node or not. It has been proven that only the failure of a boundary node may risk the biconnectivity of the network. A sufficient condition is derived to categorize the recovery process. Two versions of ARA-2C are proposed for which a decrease in the algorithm complexity is traded off for a potential increase in the total distance that the actors collectively travel. Both ARA-1C and ARA-2C are completely distributed and require only the knowledge of two-hop neighbours, which reduces the communication overhead significantly. We have validated ARA-1C and ARA-2C through simulation; comparing the performance to that of a centralized approach that considers the state of the entire network in providing a node relocation plan with the least total travel distance. The simulation results have confirmed the correctness and effectiveness of both algorithms; with close-to-optimal performance, ARA-2C is shown to outperform other applicable approaches in the literature. In the future, we plan to extend our approach to consider other metrics such as coverage and sensor-to-actor delay in determining the scope of the recovery and in selecting candidates for movement.

REFERENCES

[1] I.F. Akyildiz and I.H. Kasimoglu, “Wireless Sensor and Actor Networks: Research Challenges,” Elsevier Ad Hoc Network J., vol. 2, pp. 351-367, 2004.
[2] Udayakumar R., Khanaa V., Kaliyamurthie K.P., "High data rate for coherent optical wired communication using DSP", Indian Journal of Science and Technology, ISSN : 0974-6846, 6(S6) (2013) 4772-4776.
[3] K. Akkaya and M. Younis, “Coverage-Aware and Connectivity- Constrained Actor Positioning in Wireless Sensor and Actor Networks,” Proc. 26th IEEE Int’l Performance Computing and Comm. Conf. (IPCCC ’07), Apr. 2007.
[4] Vijayaprakash S., Langeswaran K., Gowtham Kumar S., Revathy R., Balasubramanian M.P., "Nephro-protective significance of kaempferol on mercuric chloride induced toxicity in Wistar albino rats", Biomedicine and Aging Pathology, ISSN : 2210-5220, 3(3) (2013) pp.119-124.
[5] K. Ozaki et al., “A Fault-Tolerant Model for Wireless Sensor-Actor System,” Proc. Second IEEE Int’l Workshop Heterogeneous Wireless Sensor Networks (HWISE ’06), Apr. 2006.
[6] Udayakumar R., Khanaa V., Kaliyamurthie K.P., "Optical ring architecture performance evaluation using ordinary receiver", Indian Journal of Science and Technology, ISSN : 0974-6846, 6(S6) (2013) pp. 4742-4747.
[7] P. Basu and J. Redi, “Movement Control Algorithms for Realization of Fault-Tolerant Ad Hoc Robot Networks,” IEEE Networks, vol. 18, no. 4, pp. 36-44, Aug. 2004.
[8] Sundararajan M., "Optical instrument for correlative analysis of human ECG and breathing signal", International Journal of Biomedical Engineering and Technology, ISSN : 0976 - 2965, 6(4) (2011) pp.350-362.
[9] K. Akkaya, M. Younis, and M. Bangad, “Sink Repositioning for Enhanced Performance in Wireless Sensor Networks,” Computer Networks, vol. 49, pp. 434-512, 2005.
[10] Udayakumar R., Khanaa V., Kaliyamurthie K.P., "Performance analysis of resilient ftth architecture with protection mechanism", Indian Journal of Science and Technology, ISSN : 0974-6846, 6(S6) (2013) pp. 4737-4741
[11] X. Liu, L. Xiao, A. Kreling, and Y. Liu, “Optimizing Overlay Topology by Reducing Cut Vertices,” Proc. ACM Int’l Workshop Network and Operating System Support for Digital Audio and Video (NOSSDAV ’06), May 2006.
[13] P.JENNIFER, DR. A. MUTHU KUMARAVEL, Comparative Analysis of advanced Face Recognition Techniques, International Journal of Innovative Research in Computer and Communication Engineering, ISSN(Online): 2320-9801, pp 4917-4923 Vol. 2, Issue 7, July 2014
[14] Dr.R.Udayakumar, Computer Simulation of Polyamidoamine Dendrimers and Their Complexes with Cisplatin Molecules in Water Environment, International Journal of Innovative Research in Computer and Communication, ISSN(Online): 2320-9801,pp 3729,25-30, Vol. 2, Issue 4, April 2014
[15] DR.A.Muthu kumaravel, Mr. Kannan Subramanian, Collaborative Filtering Based On Search Engine Logs, International Journal of Innovative Research in Computerand Communication Engineering, ISSN(Online): 2320-9801,pp 2432-2436, Vol. 2, Issue 1, January 2014
[16] Dr.A.Muthu Kumaravel, Mining User Profile Using Clustering FromSearch Engine Logs, International Journal of Innovative Research in Computerand Communication Engineering, ISSN(Online): 2320-9801,pp 4774-4778, Vol. 2, Issue 6, June 2014
[17] P.Kavitha, Web Data High Quality Search - No User Profiling, International Journal of Innovative Research in Computerand Communication Engineering, ISSN(Online):2320-9801,pp 2025-2030, Volume 1, Issue 9, November 2013