International Journal of Applied Engineering Research ISSN 0973-4562 Volume 10, Number 17 (2015) pp 37326-37331

Self-Stabilization of Cluster Head Mobile Ad Hoc Routing Protocolthrough the Fuzzy Relevance Degree of the Node

T Madhu
Research Scholar, JNTUK, Kakinada &Associate Professor, Department of Computer Science and Engineering, SRTIST, Nalgonda.
madhuthallapalli@yahoo.com

S S V N Sarma
Professor of CSE,Vagdevi College of Engineering,Warangal.
Ssvn.sarma@gmail.com

J V R Murthy
Professor of CSEJNTUCE, JNTUK, Kakinada.
mjonnalagadda@yahoo.com

Abstract
Inthe Mobile Ad Hoc Networks (MANETS), the integration and dis-integration of mobile nodes in and out of the clusters affects the stability of the network. Thus, a re-configuration of the cluster heads is essential which is inevitable in the traditional clustered routing protocol. This is asignificantproblembecause frequent alterations in the cluster head badlydisturb the performance of clustered routing protocols.Self-stabilization is ageneralization of fault tolerance for transient errors. Instinctively, a self-stabilizing system is a procedure that initiates from any conceivable state. In this paper, we explore the likelihood to alterarandomclustering distributed protocol into a self-stabilized clustering routing protocols. The FRCA Algorithm [7]assures a threshold that is a Fuzzy Cluster Size on the number of nodes that a cluster head can hold. Therefore, the cluster heads of the network are burdened at any time in this Algorithm. This paper introduced the self-stabilizing property into the existing fuzzy relevance based clustering algorithm as to minimize the cluster head overheads in the network and proved that the closeness and convergence of the proposed self-stabilized network is achieved using some proofs of correctness.

Keywords: self-stabilizing, Cluster Head, Fuzzy size, Mobile Ad-hoc Networks

Introduction
The self-stabilizing issue has been presented by Dijsktra [8] and ismeasured as anotion of fault tolerance, in a systemwhere transient disasters couldfraudulentinformation, messages and program counters, but not the program code. Every such failure is followed by a long period of time deprived ofany addedfailures. During this period, the model must reclaim its steadiness by itself, without any form of outdoorinterference. Therefore, a self-stabilizing system can be defined as a protocol that is initiated from any possible state. Awerbuch, Patt-Shamir and Varghese [9] considersany system self-stabilized where there is a subsection of the state set and genuine states authenticating:

- Every successor of a genuine state is genuine
- Initiated from any state, the protocolcontinuouslyultimately reaches a genuine state.

Thus, the subgroup of genuine states can be defined as the group of states accessible from some random initial states. Here, the initial state is consideredto be unique.
Self-stabilization is astrikingmethodology to with stand transient errors such as link failures or novel link formationsowing to the nodes mobility in Mobile Ad Hoc Networks. Self-stabilization [10] is auniversalarchetype to affordadvancing recovercapability to distributed systems and networks. Instinctively, a procedurefor networks or distributed algorithms are self-stabilizing if it is capable to recuperatelack ofany exteriorinterference from any disastroustemporary failure. Amongst the numerous self-stabilizing solutions existingnowadays [11], the utmostbeneficialmethods for actual networks are those that confesseffectual implementations.Maximumworks arecommitted to enhanceeffectiveness later to disasterensue, i.e., diminishing the stabilization time and the increasing thevolume of period one need to wait earlierto disasterretrieval. Self-stabilization allows the modelto recover unconventionally from arandom state, making the structurerecuperate from faults and provisionallywreckedexpectations.
The usualsurveillance of the situationstipulates that there is a need to self-stabilized computerizedmodel that is accomplished by providing the stabilization at numerous level. The self-stabilization should be achieved at program level, processor level, operating system level and network level, etc. Therefore, the author constraints his novelty in providing self-stabilization of different networks that are normally in use. In this proposed methodology, the self-stabilization is provided to the Wirelessand Mobile Adhoc Networks.
Wireless and Mobile Ad hoc Networks are infrastructure less, multiple hops, activesystemdevelopedthrough a group of mobile nodes. Wireless and Mobile Adhoc Networkscomprises of mobile nodes that comprises of the networkdeprived of immobileorganization or

integratedmanagement. In these protocols, every node converses with anothernodedirectly or throughthe intermediary nodes. This form of routing protocol is extremelyattractiveowing to the shortage of infrastructure, priceefficiency and meek installation. The contemplation in these protocols are to enhance the systemsteadiness, scalability, bandwidth utilization, and resourcedistribution and administrationcompetence.

A. Motivation

There exist a large number of clustering algorithm in the literature for wireless and mobile Adhoc networks and sensor networks based on different criteria like the highest node ID, Lowest node ID, node weights, node links, etc. Though FRCA algorithm could extremely sustain bandwidth utilization and network efficiency, minimize energy utilization, and maximize resource allotment and administration. It still faces with the problem of stabilizing the network and the routing protocol in the network. In order to address this problem and to obtain for a higher self-stabilizing network, a novel self-stabilizing methodology is proposed. In [7] proposed a fuzzy based clustering algorithm for Wireless Mobile Adhoc Sensor Networks where Cluster Heads (CH) are formed in the mobile network with fuzzy value or Fuzzy Relevance Degree (FRD) and the algorithm is named as Fuzzy Relevance-based Cluster head selection Algorithm (FRCA) that professionallygroups and accomplishes nodes by means of fuzzy information of the node position in the routing protocol.

B. Organization of the Paper

A Brief Discussion of Self-Stabilization and Motivation for the paper is given in this section. Section 2 discusses the different existing methodologies of clustering algorithm that adopts self-stabilization. The Definition of Self-Stabilization is given in Section 3. The proposed Self-Stabilized Fuzzy Relevance Based Clustering algorithm is briefly given in section 4. Self-stabilization construction and its algorithm for the Fuzzy Relevance Based Clustering Algorithm is given in section 5. Section 6 gives the proofs for the self-stabilization of the FRCA algorithm followed by section 7 which concludes the paper.

Existing Methodology

In [1] suggested a self-stabilized cluster routing methodology for MANET depending on link-cluster design. This protocol assures that beginning in arandom configuration and in limited number of stages, the system is divided into groups by means of link-cluster design.In [2] recommended the initial self-stabilizing distributed (k, r)-clustering algorithm. A deterministic procedurewarrantiesentire nodes, if conceivableaimed atspecified topology, take*k* cluster heads contained by*r* hops. An arbitraryprocedureallows the groups of cluster heads alleviate to aindigenous minimum.

In [3] presented a self-stabilizing disseminatedprocedurewhichcalculates a subcategory D of I that is a negligible k-dominating group. This methodologydepends on comparison that necessitatesO (log n) per method, congregates in O (n) circles and O (n^2) steps, where n is the magnitude of the system, and mechanismsunderneathadiscriminating scheduler.In [4]suggested a self-stabilizing type of DMAC and GDMAC algorithm. Thesemanagedby means ofan early configuration. These aresimilarlyfamiliarize to random topologicaldeviationsbecause to node crash disasters, communication link crash disasters, node recuperating or link recuperating, integration of numeroussystems, and so on.

In [5], presented a robust self-stabilizing clustering algorithm for ad hoc network. The sturdinessfeatureassurances thatinitially from arandomposition, in a single round, the system is segregated into groups or clusters. Afterward, the protocolremainssegregatedat the time of convergencein the direction of a genuine configuration where the cluster's partition is optimum.In [6] proposed a self-stabilizing clustering protocol. This protocol ensures a threshold (Size Bound) on the number of nodes that a cluster head can hold. The principle of the cluster heads selectiondepends on their weight value, a commonconsideration that arecalculatedrendering to numerous node constraints as transmission power, battery power.

The Formal Definition of Self-Stabilization

A distributed routing protocol is demonstrated as an undirected graph G = (V, E), given V is the group of nodes and E is the group of edges. An edge $(u, v) \in E$exist ifand only if u and v can straight forwardly interconnect each other. The collections of adjacent members of a node $v \in$ V representedbyN_v. At a node v in the system is allocatedadistinctive identifier (ID). For easiness, presentevery node is familiarizedwith its ID and it is indicatedusing v. We consider the nearby shared memory methodology of communication. Therefore, every node v has a limitedgroup of indigenous variables in a way that the variables at a node $v$isgiven by v and any adjacent memberof$v$, nevertheless can be solitaryalteredusing$v$.

The program forevery node v comprises of a finite group of rules. Arule is a shieldeddeclaration of the form$Rule_i$: $BGuard_i \rightarrow Act_i$, where $BGuard_i$ is a Boolean predicate concerningnative variables of$v$ and native variables of its surroundings,Act_i is a procedurewhichtransform the local variablesof v. Act_iisperformed by node $v$solitary if the $BGuard_i$ issatisfied, where it is said that node v is empowered.The position of a node is specifiedusing values of its native variables.An arrangement of a distributed model G is an illustrationof the node positions. Infinal configuration, neither of the nodes arepermitted. A calculation$e$of a system $G$is a series of configurations$co_1, co_2, c_{o3} \ldots\ldots$such that for $i = 1,2,3, \ldots\ldots$ the configuration$co_{i+1}$is attained from $co_i$by a solophase of numerous permittednodes. The nodes implement their procedures asynchronously.Thus, the sub group of permitted nodes that promptly perform their action in the course of the computation stage, is randomly preferred. A calculation is reasonableif any node inG uninterruptedlyempowered alongside a computation, will ultimatelyaccomplish a task.

The explorations are done only one reasonable and unbiasedcomputation.An estimationis bestif it achieves terminalconfiguration or it is unlimited. ConsiderCO as the group of conceivableconfigurations and εbe the group of entirely possible computationsof a system$G$. A group of calculations of Ginitiatesthrough the specificpreliminary configuration $c \in C$which is representedbyε_c. The other group of calculations of εwhose preliminary configurationsare entirely thecomponents of A$\in C$ which is denoted byε_A.The idea of attractor [13] is employed in this paper, to describe self-stabilization.

Definition 1 (Attractor): Let$A_1$and A_2be subsets ofCO. Formerly$A_3$is an attractor from A_2if:

- Convergence -$\forall e \in \varepsilon_{A_2}, (e = co_1, c_{o2}, \ldots\ldots..), \exists i \geq 1: co_i \in A$
- Closure - $\forall e \in \varepsilon_{A_3}, (e = co_1, co_2, \ldots\ldots..), \forall i \geq 1: co_i \in A_3$

Observation 1:LetA_1, $A_2$and A_3be sub groups ofCO. If $A_3$is an attractor from$A_2$and if $A$is an attractor from A_1then $A_3$is an attractor fromA_1.A group of configurations correspondingto the problem requirementsareknown as the set of genuineconfigurations, denoted as$\mathcal{L}$. $\mathrm{C}/\mathcal{L}$representsa group of illegal configurations.

Definition 2 (Self-stabilization): A disseminated system S is self-stabilized if there exists a non-empty group$\mathcal{L} \subseteq \mathrm{C}$ such that thesubsequentcircumstances handle:

1. $\mathcal{L}$is an attractor from C.
2. $\forall e \in \varepsilon_{\mathcal{L}}, e$satisfies the Specification Problem.

Self-Stabilized Cluster Head Mobile Ad Hoc Routing Protocol

A novel self-stabilizing methodology is proposed in the paper externally apart from the given clustering algorithm. A self-stabilizing network should always satisfy the closure and convergence property. This paperintroducedthe self-stabilizing property into the novel modified self-stabilized fuzzy relevance basedclustering algorithm and proves that the closeness and convergence of the proposed self-stabilized network that is achieved using some proofs of correctness.

In this paper, a Fuzzy Relevance Degree (FRD) groundedrouting protocol, i.e., FRD_v is allotted to every node where $v \in V$in the protocol. In ad hoc mobile networks, Fuzzy Relevance Degree (FRD) is a fuzzy valueμ ($0 \leq \mu \leq 1$), giventhrough an accessible power, distance, and mobility. To minimize the computational complication, a fuzzy value μ is set amongst 0 and 1 which can contain the values such as {0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1}. FRD is employed for choosing the cluster head and for building of clusters in the routing protocol. The selection of the cluster headsdepends on the FRD valuerelated to each and every node: the maximum the FRD value of a node, the healthier this node is suitably designated as acluster head. The nodes that are attaininga unique identifier (ID) can substitute the FRD value variable by the node set as (FRD value, ID), to guarantee that nodes have diverseFuzzy values. Clustering is nothing butsegregating its nodes into groups, every group with an associated cluster head and conceivably some gateway in the cluster, Cluster Members (CM) nodes as the ordinary nodes. So as to have healthysensible and functional clusters, the given stable Fuzzy clustering featuresneed to be fulfilled:

- *Eachand Every ordinary node (Gateway node and Cluster Member node)constantlyassociates with uniquecluster head of its surrounding which has maximumFuzzy Relevance Degree (FRD)compared to its FRD (Node AssociationCondition)*: This confirms that every ordinary node has straightforwardadmittance to its cluster headassociated with anappropriate cluster
- *There is at most Fuzzy Cluster Size nodes in a cluster (Cluster Size Condition)*: This guarantees that a cluster head will never be overloaded by the organizationcapability of its cluster.
- *If a cluster head v has aadjacentcluster head u such that FRD_u>FRD_v then similar size of u's cluster is Fuzzy Cluster Size (Cluster Head Neighboring Condition)*: This bounds the number of clusters. A node remains as thecluster headsolitary if it can't linkwith aprevailing cluster in its surrounding and all its appropriate clusters are occupied (i.e., they have Fuzzy Cluster Size members).

Construction of Self-Stabilization Using Fuzzy Clusters Size

The procedurefor the "self-stabilizing Cluster head routing protocol through theFuzzy Relevance degree" is given in Algorithm1 and its constants, variables, and macros are defined in Table 1.

Table1:description of Constants, Variables and Macros on Fuzzy Node v

<u>*Constants:*</u>

FRD_v:N - is the Fuzzy Relevance Degree of node v

$Fuzzy\ Cluster\ Size$: N - is the maximum limitof the cluster size

<u>*Variables of node$v$:*</u>

Chb_v:$boolean$ - specifieswhether$v$exist or not exist incluster head.

$FHead_v$: IDs - is the cluster headof$v$.

FCD_v: $\{IDs\}$ - is the list of nodes which can pick$v$ as theircluster head. If v is an ordinary node, this list should be vacant.

Sc_v:N - is the dimension of $v's$ cluster. If v is an ordinary node then $Sc_v$is 0.

<u>*Functions:*</u>

$v's$ Adjacent could be cluster headsof:

$$N_v^+ := \{z \in N_v : v \in FCD_z \wedge FCh_z = T \wedge FRD_z > FRD_{Head_z} \wedge FRD_z > FRD_v\}$$

The size of $v's$cluster:$Size_v := |\{z \in N_v : FHead_v = v\}|$

<u>*Calculationof $FCD2_v$:*</u>

Start

$$FCD2_v := \{z \in N_v:\ FRD_{Head_z} < FRD_v \wedge FRD_z < FRD_v\}$$

if $|FCD0_v| \leq Fuzzy\ Cluster\ Size - Size_v$ then $FCD1_v \coloneqq FCD0_v$
else $FCD1_v$ entails the $Fuzzy\ Cluster\ Size - Size_v$ smallest member of $FCD0_v$
if $FCD_v \subseteq FCD1_v \cup \{FRD \in N_v : FHead_{FRD} = v\}$ then $FCD2_v \coloneqq FCD1_v$
else $FCD2_v \coloneqq \emptyset$
Stop

Notation 1:

- $Cluster_v$ is the group of nodes pertaining to the $v's$ cluster (partakingelected v as their cluster head): $Cluster_v := z \in N_v : FHead_z = v$
- The security predicate is given as: $P_s(v) \equiv |CD_v \cup Cluster_v| \leq Fuzzy\ Cluster\ Size$

Algorithm 1:Self-Stabilizing Cluster Head Routing Protocol through the Fuzzy Relevance Degree on Node v

<u>Predicates:</u>

$$G_0(v) \equiv [(S_{Head_v} > Fuzzy\ Cluster\ Size) \vee (FHead_v \notin N_v) \vee (FCH_{Head_v} = F) \vee (FRD_{Head_v} < FRD_v)$$
$$G_{11}(v) \equiv (FCh_v = F) \wedge N_v^+ = \emptyset \wedge G_0(v)$$
$$G_{12}(v) \equiv (FCh_v = T) \wedge N_v^+ = \emptyset \wedge Head_v \neq v$$
$$G_1(v) = G_{11}(v) \vee G_{12}(v)$$
$$G_2(v) \equiv N_v^+ \neq \emptyset$$
$$G_3(v) \equiv (FCh_v = T) \wedge [(Sc_v \neq Size_v) \vee (FCD_v \neq FCD2_v)]$$
$$G_4(v) \equiv (FCh_v = F) \wedge [(Sc_v \neq 0) \vee (FCD_v \neq \emptyset)]$$

<u>Rules</u>

$$R_1(v): G_1(v) \to FCh_v \coloneqq T;\ Sc_v \coloneqq Size_v;\ FCD_v \coloneqq FCD2_v;\ FHead_v \coloneqq v$$
$$R_2(v): G_2(v) \to FCh_v \coloneqq F;\ Sc_v \coloneqq 0;\ FCD_v \coloneqq \emptyset;\ FHead_v \coloneqq max_{FRD_z}\{z \in N_v^+\}$$
$$R_3(v): \neg G_1(v) \wedge \neg G_2(v) \wedge \neg G_3(v) \to Sc_v \coloneqq Size_v;\ FCD_v \coloneqq FCD2_v$$
$$R_4(v): \neg G_1(v) \wedge \neg G_2(v) \wedge \neg G_4(v) \to Sc_v \coloneqq 0;\ FCD_v \coloneqq \emptyset$$

We fragment the possible cases where a node $v$considers to alter atthe indigenous variables conferring to the subsequent mutually exclusivecases:

- $v$has to be a cluster head (rule$R_1(v)$). In $v's$ surrounding, there is no appropriatecluster head (i.e.,$N_v^+$is vacant) and it pertains to a cluster that has more than $Fuzzy\ Cluster\ Size$ nodes or v does not satisfy the Affiliationcondition. In this case, $G_{11}(v)$ is satisfied.
- $v$has to changetocluster (rule$R_2(v)$). The group ofN_v^+ is not vacant: v has in its surrounding more appropriatecluster headcompared to its current one. In this case, $G_2(v)$is satisfied.
- $v$has to altercertain local variable values without altering of cluster. If v is a cluster head then $G_3(v)$ or$G_{12}(v)$ is satisfied. If v is an ordinary node then $G_4(v)$ issatisfied.

Proof of Self-Stabilization

In subsection 7.1, it is shown that the safety predicate graspsuninterruptedlyuntil the routing protocol converges to a genuine configuration, as soon as the configuration accomplishes this predicate. Primarily allclusters considers less than n (varies for different clustered routing protocol) members, this characteristic is provedlengthwise any calculationaccomplishing a genuine configuration. A genuine configuration is a final configuration whereeach and every node v fulfills$P_s(v)$and$FCD_v \coloneqq \emptyset$. In subsection B, it is shown that along any reasonablecalculation, a genuineconfiguration is attained from any configuration. In subsection C, it is considered that in a genuine configuration is established which satisfies the well-adjusted clustering properties.

A. Security

*Definition 3:*Consider$A_2$to be the group of secure configurations definedby $\{C | \forall v: P_s(v) \text{is satisfied}\}$

Notation 2:Consider$c$to be a configuration. $FCD_v(c)$representsas the value of the FCD variable for the node $v$in c. $Cluster_v(c)$is represented as the $v's$ cluster in Configuration$c$.

<u>*Observation 2*</u>:consider that there is aestimationstep$co_1 \overset{cs}{\to} co_2$. Rendering to the functionN^+ and to the ruleR_2,$Cluster_v(co_2) \subseteq Cluster_v(co_1) \cup FCD_v(c_1)$
Consider that $v$always keep informedFCD_vat the time of calculationstepcs. Rendering to the function $FCD2_v$,

- $FCD_v(c_{o2})= \emptyset$ or$|FCD_v(c_2) \cup Cluster_v(co_1)| \leq Fuzzy\ Cluster\ Size$
- $FCD1_v(co_1) \cap Cluster_v(co_1) = \emptyset$.

Lemma 1**:**A_2 is closed.

<u>Proof:</u>Considerthat: (1) there is a configuration $co_1$where$P_s(v)$ holds and (2) partake a calculationstep$co_1 \overset{cs}{\to} co_2$. It is showed that $P_s(v)$graspsinc_2.

In any case ($FCD_v(co_2)= FCD_v(co_1)$, $FCD_v(co_2) =\emptyset$, $FCD_v(co_2) =FCD2_v(co_1)$), we have$|FCD_v(co_1) \cup Cluster_v(co_1)| \leq Fuzzy\ Cluster\ Size$ and$|FCD_v(co_2) \cup Cluster_v(co_1)| \leq Fuzzy\ Cluster\ Size$.

<u>Case1</u>: $Cluster_v(co_2) \subseteq Cluster_v(co_1)$. $P_s(v)$is satisfiedin co_2.

<u>Case2</u>: $Cluster_v(co_2) \subseteq \{u_1, u_2, \ldots\ldots u_m\} \cup Cluster_v(co_1)$.

Rendering to the observation 2, $\{u_1, u_2, \ldots\ldots u_m\} \cup Cluster_v(c_1)$
If $FCD_v(co_2) = FCD_v(co_1)$ or $FCD_v(co_2) = \emptyset$ then$|FCD_v(co_2) \cup Cluster_v(co_2)| \leq |FCD_v(co_1) \cup \{u_1, u_2, \ldots\ldots u_m\} \cup Cluster_v(co_1)| \leq |FCD_v(co_1) \cup Cluster_v(co_1)| \leq Fuzzy\ Cluster\ Size$.
$P_s(v)$is satisfied in co_2

If $FCD_v(co_2) = FCD2_v(co_1)$ then (according to the $FCD2_v$ definition) $FCD_v(co_1) \subseteq (FCD2_v(co_2) \cup Cluster_v(co_1))$

Thus$|FCD_v(co_2) \cup Cluster_v(co_2)| = |FCD_v(co_1) \cup \{u_1, u_2, \ldots \ldots u_m\} \cup Cluster_v(co_1)| \leq |FCD_v(co_1) \cup Cluster_v(co_1)| \leq Fuzzy\ Cluster\ Size$.
$P_s(v)$is satisfied inco_2

B. Convergence

Some of the proofs are given in [12] that are not mentioned in this section.

Lemma 2:$P_g \equiv (G_{12}(v) = F) \wedge G_4(v) = F)$.

$A_4 = A_2 \cap \{c \in C | \forall v: FHead_v \in N_v \cup \{v\}And|Cluster_v| \leq Fuzzy\ Cluster\ Size$and $P_g(v)$is satisfied$\}$ is an attractor from$C$.

The convergence is completed in stage. At the termination of the i^{th}phaseconfiguration set $CL''i$ is attained: entire nodes of set_ihas selected their cluster head. We define set_i and $CL''i$ as given.

Notation 3: $CL''0 = A_4$ and $set_0 = \emptyset$.$V_i$is the set of nodes that do not pertain to set_i: $V_i = V - set_i$.$vh_i$is the node with the maximum weight in V_i.

- $CL_{i+1} = CL''i \cap \{c \in C | FCh_{vh_i} = T\}$. $Fuzzy\ Cluster\ Size_i$ is the value min ($Fuzzy\ Cluster\ Size, |N_{vh_i} \cap V_i|$)
- $CL\grave{}_{i+1}$=$CL_{i+1} \cap \{c \in C | |Cluster_{vh_i}| = Fuzzy\ Cluster\ Size_i\}$
- $set_{i+1} = set_i \cup \{vh_i\} \cup Cluster_{vh_i}$
- $CL''i + 1 = CL\grave{}_{i+1} \cap \{c \in C \mid \forall v \in set_{i+1} : FCD_v = \emptyset\}$.

***Observation 3*:**Consider v be a node ofV_i then, by definitionofV_i, $FHead_v \notin Set_i = V\ then\ set_i \subset set_{i+1} and\ set_i = set_{i+1}$

At each stage, set_iupsurges till it containsentire nodes. Onceset_i= , we prove that authentic configuration is attained.

Lemma 3: For any value of i, CL_{i+1} is an attractor from C,assuming that $CL''i$ is an attractor from C.

Proof: vh_ibe the node with the maximum Fuzzy Relevance Degreein V_i. Considerz to be in N_{vh_i}. If $z \in set_i$, the $FCD_z = \emptyset$. So $N^+_{vh_i}$is empty: vh_i never executes$R_2(vh_i)$.

According to the observation 3, consider$FHead_{vh_i} \in V_i$,therefore by definition of vh_i, $FRD_{Head_{vh_i}} \leq vh_i$. If vh_i is not a cluster headformerly$G_1(vh_i)$is satisfied because$FRD_{Head_{vh_i}} \leq FRD_{vh_i}$. As all computations are reasonable, vh_ieventually performs$R_1(vh_i)$. After that FCH_{vh_i} = T and$FHead_{vh_i} = vh_i$, continually.

Lemma 4:consider oc_1 be a configuration ofCL_{i+1}. Let cs be acalculationstagefromco_1:$co_1 \xrightarrow{cs} co_2$. We have$Cluster_{vh_i}(co_1) \subseteq Cluster_{vh_i}(co_2)$

Proof: Assume that $u$is a node of $Cluster_{vh_i}$ (i.e.,$FHead_u = vh_i$). InCL_{i+1}, anadjacent z of u such that $FRD_z > FRD_{vh_i}$inset_i:$FCD_z = \emptyset$. Thus $z \notin N^+_u$; we accomplish that N^+_u is emptyalways. Therefore, $G_2(u)$ is never proved. In CL_{i+1},$G_1(u)$ isnot proved. We accomplish that the node u remains in the Clusterof vh_icontinually.

Lemma 5:considerc_1 be a configuration ofCL_{i+1}. $cs$is a calculationstage from co_1:$c_1 \xrightarrow{cs} co_2$. We have$FCD1_{vh_i}(co_2) = FCD1_{vh_i}(co_1)$if only if $Cluster_{vh_i}(co_1) = Cluster_{vh_i}(co_2)$

Lemma 6: For any value of$i$, $CL\grave{}_{i+1}$ is an attractor from Cpresuming that $CL_{i+1}$is an attractor from C.

Proof: Once CL_{i+1} is attained, solitary the nodes of $V_i$might bein the Cluster ofvh_i, consequently, the size of $Cluster_{vh_i}$ isrestrictedby$Fuzzy\ Cluster\ Size_i$. Since no node can abandon from$Cluster_{vh_i}$(lemma 4), $Cluster_{vh_i}$will ultimatelyremainindistinguishablealways.
Conferringto lemma 5,$CD1_{vh_i}$ will finallyremainas indistinguishablecontinually.Once $FCD1_{vh_i}$ is computed, if, FCD_{vh_i} =$FCD1_{vh_i}$, $R_3(vh_i)$ isnot everpermitted (i.e., FCD_{vh_i}remainsconstantly equal to $CD1_{vh_i}$).Once, $FCD1_{vh_i}$ is computed, if FCD_{vh_i}= $FCD1_{vh_i}$ then$R_3(vh_i)$ is permittedpersistently. By equality, $R_3(vh_i)$ actionwill be ultimatelyaccomplished. Afterwards at most two $R_3(vh_i)$actions, we consideredFCD_{vh_i}= $FCD1_{vh_i}$. We accomplish thatany computation has a suffix where FCD_{vh_i}remain equalto $FCD1_{vh_i}$. In this suffix, the size of $FCD1_{vh_i}$ is equivalent to$Fuzzy\ Cluster\ Size_i - |Cluster_{vh_i}|$

Consider that the magnitude of $Cluster_{vh_i}$ is persistentlylessthan$Fuzzy\ Cluster\ Size_i$. In that case, the computation of a node u will remaincontinually in the groupFCD_{vh_i}. Bydescription of vh_iandV_i, we have$FRD_{Head_u} \leq FRD_{vh_i}$ and$FRD_u < FRD_{vh_i}$, therefore$vh_i \in N^+_u$;.Any neighbor z of u such that$FRD_z > FRD_{vh_i}$ is inSet_i. Thus $FCD_z = \emptyset$. Therefore vh_i is thenode of N^+_u having the highest Fuzzy Relevance Degree. $R_2(u)$ispermittedpersistently. By equality, $R_2(u)$ action will be ultimatelyachieved:u will selectvh_i as its clusterhead: $Cluster_{vh_i}$isaltered. There is aninconsistency.

Lemma 7: For any value of i, CL''_{i+1} is an attractor from C,assuming that $CL\grave{}_{i+1}$ is an attractor from C.

Proof: Let v be a node of Set_{i+1} that does not pertain toSet_i. If v is an ordinary node, $FCD_v = \emptyset$ because A_3 isa subset of CL''_{i+1}. If v is a clusterhead then $v = vh_i$.
By definition of $CL\grave{}_{i+1}$, $\forall u \in N_v$: $FRD_{Head_u} \leq FRD_v$ or$|Cluster_v|$ = $Fuzzy\ Cluster\ Size$. Thus $FCD2_v = \emptyset$, in $CL\grave{}_{i+1}$.

If $FCD_v = \emptyset$, then $R_3(u)$ is permittedpersistently. Once the ruleis implemented, $CD_v = \emptyset$ holds.

Theorem 1:consider$A_5$to be a configurations group defined by $A_5 = A_4 \cap \{c \in C \mid \forall v : CD_v = \emptyset\}$. The protocol ultimately attains a terminal configuration ofA_5.

Proof: Rendering to the Observation 3, $set_i \subset set_{i+1}$.Consequently, there exists j such that set_j= .$CL''j$is an attractor because $CL''0$, L_i, $CL\grave{}_i$, and $CL''i$ are attractorsfor any value of $1 \geq i \leq j$. In $CL''j$, the rule R_1, R_2, and R_4 are not permitted on any node. Solitary therule R_3might be allowedendlessly, on a node$v$. By equality,v will implement$R_3(v)$, then v is not everpermitted. It isaccomplishthat a terminal configuration of CL''_i will be attained. Anyconfiguration of CL''_i belong to A_5.

C. Correctness

Theorem 2: If the terminal configuration of A_5 isattained then the well-adjusted clustering properties are satisfied.

Proof: In the terminal configuration ofA_5, for every node$z$,$G_i(z) = F : i = 1..4$ and $FCD_z = \emptyset$is considered(see theorem1).

Case 1:z is an ordinary node. $G_{11}(z)$ = F implies($FHead_z \in N_v$) $\wedge$ (FCH_{Head_z} = T) and ($FRD_{Head_z} > FRD_z$).Thus, z satisfies affiliation condition.

Case2:z is a cluster head node. Following Lemma 2, ina terminal configuration $S_v \leq Fuzzy\ Cluster\ Size$, thus, the sizecondition is satisfied.Let v be a clusterhead, neighbor of z such that $FRD_v > FRD_z$.Notice that $FCD0_v$ is not empty (it contains z). $G_3(v)$isnot verified. Thus, $FCD2_v$ is equal to FCD_v. We have$FCD_v = \emptyset$, thus, $FCD2_v = CD1_v = \emptyset$. We concludethat $Size_v$= $Fuzzy\ Cluster\ Size$. Thus, every clusterhead v in z'neighborhood authenticates the given predicate: ($FRD_v \leq FRD_z$)or (Sc_v= $Fuzzy\ Cluster\ Size$). Thus, the clusterhead neighboringcondition is fulfilled.

Conclusions

As high speed systemsare increasing higher and higher, it is important to developstraight forward cluster routingstructures with a comparativelytrivial memory prerequisite.In this paper, a self-stabilized cluster head routing algorithm for MANET is given based on Fuzzy Relevance Degree (FRD) design. The proposed methodology greatly improved the stability of the clustering scheme bypersisting the worth of clustering algorithm and the overload of the cluster head. In this proposed methodology, the performance of clusters depends on the values of Fuzzy Relevance Degree and Fuzzy Cluster Size. A deterministic procedureassurances that allnodes, if conceivable for the given topology, have efficient cluster's with fuzzy cluster size and achieves a stabilized routing protocol.

References

[1] DoinaBein, Ajoy K. Datta, Chakradhar R. Jagganagari, Vincent Villain, "A Self-stabilizing Link-Cluster Algorithm in Mobile Ad Hoc Networks", Proceedings of the 8th International Symposium on Parallel Architectures, Algorithms and Networks (ISPAN'05), IEEE, 2005.

[2] Andreas Larsson, PhilippasTsigas, "A Self-stabilizing (k,r)-clustering Algorithm with Multiple Paths for Wireless Ad-hoc Networks", IEEE, 2011.

[3] Ajoy K. Datta, Stephane Devismes, Lawrence L. Larmore, "A Self-Stabilizing O(n)-Round k-Clustering Algorithm", IEEE, 2009.

[4] Colette Johnen, Le Huy Nguyen, « Self-Stabilizing weight-based Clustering Algorithm for Ad hoc sensor Networks", IEEE, 2006.

[5] Colette Johnen, Le Huy Nguyen, « Robust Self-Stabilizing Clustering Algorithm", IEEE, 2006.

[6] Colette Johnen, Le Huy Nguyen, "Self-Stabilizing construction of Bounded Size Clusters", IEEE, 2008.

[7] Chongdeuk Lee and TaegwonJeong, "FRCA: A Fuzzy Relevance-Based Cluster Head Selection Algorithm for Wireless Mobile Ad-Hoc Sensor Networks", Sensors 2011, 11(5), 5383-5401.

[8] E. Dijkstra. Self-stabilizing systems in spite of distributed control. Communications of the Association of the Computing Machinery, 17:643–644, 1974.

[9] B. Awerbuch and D. Peleg. Sparse partitions (extended abstract). In the 31st Annual IEEE Symposium on Foundations of Computer Science (FOCS90), pages 503–513, 1990.

[10] E. W. Dijkstra, "Self-stabilizing systems in spite of distributed control." Communication ACM, vol. 17, no. 11, pp. 643–644, 1974.

[11] S. Dolev, Self-stabilization. MIT Press, March 2000.

[12] C. Johnen and L. Nguyen. Self-stabilizing bounded size clustering algorithm. Technical Report 1464, L.R.I, 2006.

[13] C. Johnen and S. Tixeuil. Route preserving stabilization. In the 6th International Symposium on Self-stabilizing System (SSS'03), Springer LNCS 2704, pages 184–198, 2003.

[14] Olivier Flauzac, Bachar Salim Haggar and FlorentNolot, "Self-stabilizing Clustering Algorithm for Ad Hoc Networks", 2009 Fifth International Conference on Wireless and Mobile Communications, IEEE