U.P.B. Sci. Bull., Series C, Vol. 74, Iss. 3, 2012 ISSN 1454-234x

OPTIMIZED SELECTION OF CLUSTER-HEADS IN THE FRAMEWORK OF NEXT GENERATION VEHICULAR NETWORKS

Adrian MATEI[1]

Lucrarea defineşte o nouă metrică optimizată pentru selecţia de noduri dominante, folosită la implementarea algoritmilor de grupare a nodurilor vehiculare în structuri stabile, în cadrul funcţional al Reţelelor Ad-Hoc Vehiculare de Nouă Generaţie. Pe baza metricii complexe, a fost introdus un nou algoritm de grupare, denumit OSCAR (Optimized Selection of Clusterheads AlgoRithm). Simulările au fost efectuate într-un mediu de simulare compus din NS-3 şi VanetMobiSim, pentru scenarii de tip autostradă şi metropolitan. Rezultatele obţinute arată performanţe superioare ale algoritmului OSCAR comparativ cu algoritmul DBC (Density Based Clustering) folosit în comparaţie, cu o creştere semnificativă de stabilitate a structurilor.

This paper focuses on the definition of a new optimized selection metric for the clustering of vehicular nodes, in the framework of Next Generation Vehicular ad-hoc Networks. Based on this complex metric, a new clustering algorithm was introduced, entitled OSCAR (Optimized Selection of Cluster-heads AlgoRithm). The tests were performed in an NS-3 and VanetMobiSim simulation environment for highway and metropolitan scenarios. Results obtained show that OSCAR performs better than the DBC (Density Based Clustering) algorithm used as benchmark, with a significant increase in cluster stability.

Keywords: Next Generation Vehicular Networks, clustering algorithm, cluster-head selection metric, mobility information, cluster stability

1. Introduction

Vehicular Ad-Hoc Networks (VANETs) are dynamic networks, characterized by a high degree of nodes mobility, interacting at multiple levels and moving in constrained topological environments. The high mobility causes network fragmentation and frequent disconnections, rendering the communication intermittent. In order to integrate the large variety of existing concepts and scientific contributions in the field of vehicular networks, in [1] and [2] the authors defined a new framework for Next Generation Vehicular Networks (NGVN) and a new software-oriented architectural node model, which is both

[1] PhD student, Faculty of Electronics, Telecommunications and Information Technology, University POLITEHNICA of Bucharest, Roamania, e-mail: adrmatei@gmail.com

scalable and modular, combining concepts of vehicular communications, Delay- or Disruption-Tolerant Networking (DTN) and context awareness and adaptation. This new architecture is built as an overlay on top of three levels - Mobility, Connectivity and Application, together with a cross-layering component for transversal context information processing and transfer.

One of the major challenges in designing a vehicular network is the system scalability. The vehicular network should function in parameters for both low and high node density, regardless of the topology – i.e. metropolitan, highway etc. The size of the network is an important parameter depending on the degree of mobility and on the limitations imposed by the environment. In order to reduce the size of vehicular networks, stable groups of nodes (clusters) must be identified, comprising vehicles with similar mobility patterns. The grouping of nodes in stable structures is achieved by applying clustering algorithms, built around the exchange of mobility context information between neighboring nodes. Such algorithms are implemented at the Mobility Layer in the NGVN framework defined in [2] and the implementation is based on Low Order Modules (LOMs), integrated in the software architecture defined in [1].

Clustering has been carefully studied within the academic community [3]-[9]. While some approaches are focused on the process of periodically creating stable clusters, others target the mechanisms for mobility management. In terms of mobility, the node definition is based on GPS position information (latitude and longitude), assumed to be available with most vehicles (either embedded or not), the speed of movement, the angular direction giving the movement trajectory and the topology (a map with roads, traffic signs, traffic lights, etc.). The obtained clusters exhibit the dominance characteristic, in the sense that every cluster member has direct connectivity with the Cluster-Head (CH) node. Also, CH nodes are independent, i.e. they cannot be direct neighbors, as they would then compete for dominance. All cluster members are at most at two-hop distance away from each other, due to the usage of CH as transit node.

In spite of the large variety of existing clustering schemes, most of them are derived from a set of base algorithms: Lowest-id, weighted or non-weighted Highest-connectivity (degree), Least Cluster Change (LCC) and Distributed Mobility-Adaptive Clustering (DMAC). In Highest-connectivity, the node with the largest number of neighbors at one hop away is selected as CH. This algorithm is prone to a large number of CH changes due to high mobility of nodes. Lowest-id is based on assignment of a distinct id to each node, the node with the lowest id in a neighborhood becoming CH. This algorithm outperforms the Highest-connectivity algorithm, but does not take into account mobility information. With LCC, CH change occurs only if two CH nodes come within transmission range of each other and when a node becomes completely disconnected from any other cluster. LCC outperforms both Lowest-id and Highest-connectivity, but it still overlooks the usage of the critical mobility information. DMAC overcomes this issue by adding a generic weight to each node, which maximized within a neighborhood results in the identification of the CH, thus adapting to nodes mobility. DMAC outperforms the previously described clustering methods, creating stable structures with reduced number of CH changes.

In [3] the authors introduced Density Based Clustering (DBC), an algorithm performing cluster formation based on a clustering metric that takes into account the density of the connection graph and link quality. DBC shows increased performance compared to DMAC, however CH selection does not take into account critical vehicular parameters like disruptive forwarding capabilities, node storage capacity, complex reputation and availability for custody transfers etc. In order to improve the grouping efficiency in dynamic vehicular environments, characterized by frequent disconnections, the author is proposing in this paper a new clustering algorithm – *Optimized Selection of Cluster-heads AlgoRithm (OSCAR)*. This algorithm is an extension of the Density Based Clustering (DBC) algorithm proposed in [3], using a new complex optimized selection metric for the selection of cluster-head nodes. Simulations performed in NS-3 based on mobility traces generated with VanetMobiSim for metropolitan and highway scenarios show a significant performance improvement compared to DBC. The purpose of this paper is the design of a new clustering logic, which can be combined with a routing mechanism to allow efficient data dissemination in vehicular networks.

The rest of the paper is organized as follows. In Section 2, the architectural model for Next Generation Vehicular Networks is briefly described, in order to emphasize the integration of OSCAR within the lower Mobility Layer. The next section sets up the generic terminology, as pre-requisite for building the complex CH selection metric. The metric is defined in Section 4, based on which the main threads of the OSCAR algorithm are introduced further on, in Section 5. The results of the simulations are presented in Section 6 and finally, the conclusions are listed in Section 7.

2. Architectural model for Next Generation Vehicular Networks

The architectural model, introduced in [1], is based on the framework for Next Generation Vehicular Networks (NGVN) presented in [2]. It describes the logical architecture of the Abstract vehicular Node (AN), using a two-tiered software-oriented hierarchical model based on High Order Modules (HOM) and Low Order Modules (LOM), as shown in Fig. 1. There are three HOMs, fixed from a functional point of view, implementing basic roles, specifically the Mobility Context Module (MCM), the Connectivity Context Module (CCM) and the Application Context Module (ACM). The transversal integration of these modules is provided by a Cross-Layer Context Module (CLCM). The MCM is responsible for network topology discovery, clustering mechanisms for identifying stable structures of nodes, nodes virtualization based on position, speed, angular movement direction and others. At the CCM, basic functions like routing, forwarding and custody transfers (specific to intermittent communications) are integrated. The ACM implements the addressing model and set of basic representative services, while at the CLCM specific mechanisms are activated, like scheduling, security, storage, processing of Context Information (CI), as well as convergence layers for integration with existing telecom infrastructures. The exchange of CI is based on the specific Context Managers (CM) at each HOM module and CI processing is done via the Translation Logic (TL) at CLCM.

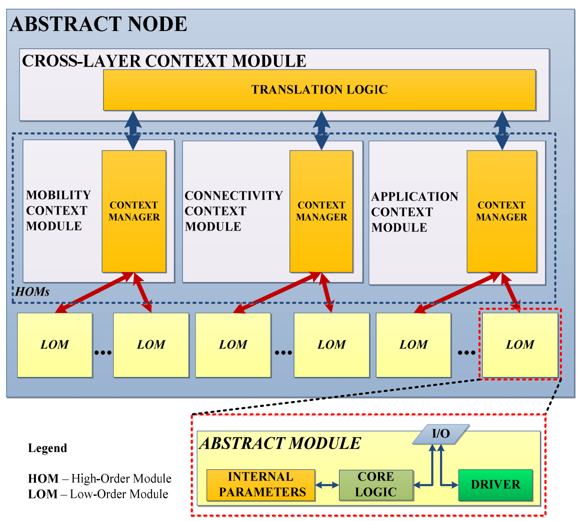


Fig. 1. NGVN architectural model for AN

Low Order Modules are used to implement specific mechanisms or algorithms at different layers of the architecture, providing the tools for integration of specific contributions proposed by the academic community – i.e. particular clustering algorithms, routing and forwarding schemes, specific schedulers, etc. Clustering algorithms are integrated as specific LOMs in the logic of the Mobility Context Module (MCM). As such, the new OSCAR algorithm proposed in this paper was defined in the framework of an LOM, using specific control parameters provided transversally by the CLCM module.

3. Generic terminology and OSCAR selection metric definition

In order to describe the mobility CI, the following notations are introduced, valid for a specific moment in time t and for a specific vehicular node n under assessment. This terminology will be used for the description of the OSCAR algorithm.

- $\mathcal{V}^{(n)}(t)$ - current neighborhood, containing a set of nodes characterized by various mobility parameters, all of them interacting with n; $NC^{(n)}(t)=\left|\mathcal{V}^{(n)}(t)\right|$ is the degree of connectivity;
- $\mathbf{p}^{(n)}(t)$, $\mathbf{v}^{(n)}(t)$, $\boldsymbol{\theta}^{(n)}(t)$ - position, speed and angular direction vectors;
- $EID(n)$ is the Endpoint Identifier (EID) address, $w^{(n)}(t)$ is the node weight, $CH^{(n)}(t)$ is the corresponding cluster-head, $w_{CH}^{(n)}(t)$ and $EID_{CH}^{(n)}(t)$ are the weight and EID of the cluster-head assigned to n.

In order to describe the intermittent capabilities of the node under assessment, the following parameters are used: $dtn(n)$ indicates the DTN capabilities; $repo(n)$ indicates the storage capabilities of bundle messages; $ste^{(n)}(t)$ describes the store-carry-forward capabilities of the node. Parameters like $dtn(n)$, $repo(n)$ are transferred by the CLCM module to the MCM module, as defined in [1]. The typology impacts the applications that can be setup, like DTN applications, SMS messages or continuous sessions. R defines the radius of the cellular area around the node where communication is possible (connectivity radius or distance). $tip^{(n)}(t)\in\{nV,nC,nM\}$ is the typology of the node, showing that the node is either a visitor, a candidate or a member and $da^{(n)}(t)$ is the lifetime during which the node is assigned to a specific cluster.

For every neighbor k of node n, $\forall k\in\mathcal{V}^{(n)}(t)$ the following information is available for processing with similar definitions as per above: EID address, neighbor weight, position, speed, angular direction. Additionally, $ts^{(n)}(k)$ is the timestamp for receiving the last beacon from neighbor k, while $nb^{(n)}(k)$ is the number of beacons exchanged between n and k. $lq^{(n,k)}(t)$ is the quality of the link between k and n defined as the link SNR averaged over a specific time interval and $dc^{(n,k)}(t)$ is the connectivity duration between k and n. The usage of the link quality parameter $lq^{(n,k)}(t)$ in the clustering process is necessary but not sufficient, due to the fluctuations on the radio channel introducing variations in the SNR and impacting cluster stability. $dc^{(n,k)}(t)$ adds extra significance to $lq^{(n,k)}(t)$. It is the positive root of the 2nd degree equation in variable τ :

$$\left[\Delta \mathbf{v}(n,k)\right]^2 \tau^2 + \left[2 \cdot \Delta \mathbf{v}(n,k) \cdot \Delta \mathbf{p}(n,k)\right]\tau + \left[\Delta \mathbf{p}(n,k)\right]^2 - R^2 = 0, \tag{1}$$

where $\Delta \mathbf{v}(n,k) = \mathbf{v}^{(k)} - \mathbf{v}^{(n)}$ and $\Delta \mathbf{p}(n,k) = \mathbf{p}^{(k)} - \mathbf{p}^{(n)}$ for a specific neighbor k of node n. Considering the degree of stable connectivity of node n in equation (2):

$$NC_{st}^{(n)}(t) = \left|\mathcal{V}_{st}^{(n)}(t)\right|, \mathcal{V}_{st}^{(n)}(t) \leftarrow \mathcal{V}^{(n)}(t)\Big|_{lq^{(n,k)}(t) \geq lq_{thr},\, dc^{(n,k)}(t) \geq dc_{thr}}, \tag{2}$$

referring to links where the quality exceeds a specific threshold. In order to increase cluster stability, a dedicated parameter is defined

$$\mathcal{H}^{(n)}(t) = \left\{\mathcal{V}_{st}^{(n)}(\tau) \middle| \tau \in \overline{t - N + 1 \ldots t}\right\}, \tag{3}$$

representing the history of stable neighborhoods $\mathcal{V}_{st}^{(n)}(t)$ for node n, for different time samples prior to moment t.

In [3] the authors introduced a re-selection metric for the cluster-head node, maximizing the node weight parameter, which is built by taking into account the current stable links and stable links from the past (based on history), as well as the stable connectivity level $NC_{st}^{(n)}$. The metric defined in [3] is effective, but can be further extended to increase the performance of the clustering algorithm.

In this paper the author is proposing **a new mathematical model for a complex cluster-head selection metric** $SM^{(n)}(t)$, derived from the DBC metric by adding additional criteria for cluster-head selection:

$$SM^{(n)}(t) = \sum_i w_i(t) \cdot \Pi_i^{(n)}(t). \tag{4}$$

In the above definition, $\Pi_i^{(n)}(t)$ represent the various **CH selection parameters** and $w_i(t)$ their corresponding weights. The CH selection is based on:

$$CH_{nou}^{(n)}(t) = k,\ k \in \mathcal{V}^{(n)}(t),\ SM^{(n)}(t) = \max,\ SM^{(k)}(t) > SM^{(CH_{curent})}(t) + \delta_{SM}. \tag{5}$$

Compared to DBC algorithm in [3], the new proposed metric $SM^{(n)}(t)$ replaces the node weight $w^{(n)}(t)$. The number of selection parameters shows the complexity of the algorithm and the flexibility of the proposed solution. Weights can be chosen to model specific behavior depending on the environment and the characteristics of the applications to be supported by the nodes. In order to avoid numerical oscillations, exponential „smoothing" is used for all parameters, with a weighting factor $\lambda \in [0,1]$, usually set to be 0.7.

The new metric for the selection of CH nodes in the OSCAR algorithm is defined by the following selection parameters:

- **Degree of stable connectivity** – obtained by counting the "stable" neighbors, current and previous, $\mathcal{V}_{st}^{(n)}(\tau) \cap \mathcal{V}^{(n)}(t)$ for all stable neighborhoods stored in the history $\forall \mathcal{V}_{st}^{(n)} \in \mathcal{H}^{(n)}(t)$ normed to the current degree of connectivity $NC^{(n)}(t)$ and the size of the history *N*:

$$\Pi_1^{(n)}(t) = \frac{\sum_{\tau=t-N+1}^{t} \left| \mathcal{V}_{st}^{(n)}(\tau) \cap \mathcal{V}^{(n)}(t) \right|}{N \cdot NC^{(n)}(t)}, \ \forall \mathcal{V}_{st}^{(n)} \in \mathcal{H}^{(n)}(t). \tag{6}$$

- **Node profile** – models the DTN forwarding capabilities, successful DTN custody transfers and node storage capabilities:

$$\Pi_2^{(n)}(t) = \frac{dtn(n) + repo(n) + ste^{(n)}(t)}{2 + ste^{(n)}(t)}, \tag{7}$$

where $dtn(n)$ and $repo(n)$ are binary values $\{0;1\}$. If there exists at least one DTN forwarding scheme implemented through an LOM, then $dtn(n) = 1$. If the storage capacity at the CLCM exceeds a minimal threshold, then $repo(n) = 1$. $ste^{(n)}(t)$ describes the DTN custody transfers already performed during the history timeframe $\mathcal{H}^{(n)}(t)$.

- **Node reputation and availability**

$$\Pi_3^{(n)}(t) = R^{(n)}(t) \cdot D^{(n)}(t), \tag{8}$$

Node availability indicates the nodes flexibility for becoming cluster-heads and working as relays for intra-cluster messages, as well as gateways for inter-cluster messages. The reputation allows specific nodes to discard bundles received from other nodes that are not “trustworthy”.

- **Spatial and temporal dependency, mobility pattern**

The node mobility pattern can be modeled by using a generic operator $\mathcal{M}\{\bullet\}$ applied to the three key vectors showing the vehicle motion, i.e. position, speed and angular direction. In the case of the speed vector, $\mathbf{v}^{(n)}(t + \Delta t) = \mathcal{M}_{\Delta t}\left\{\mathbf{v}^{(n)}(t)\right\}$. In order to highlight the spatial dependency, let us define the degree of spatial correlation between two neighboring nodes *n* and *k* using the cosine of the angle between the two corresponding speed vectors $\varphi^{(n,k)}(t) = \angle\left(\mathbf{v}^{(n)}(t); \mathbf{v}^{(k)}(t)\right)$, weighted using a min-max factor. The spatial correlation is null for any two nodes *n* and *k* placed outside the connectivity radius *R*. This is modeled using $\Delta d_e^{(n,k)}(t)$ - the Euclidian distance between *n* and *k*.

$$\delta_{sp}^{(n,k)}(t) = \begin{cases} \cos\varphi^{(n,k)}(t) \dfrac{\min}{\max}\left\{\left\|\mathbf{v}^{(n)}(t)\right\|; \left\|\mathbf{v}^{(k)}(t)\right\|\right\} \\ 0, \text{ for } \Delta d_e^{(n,k)}(t) > c_1 \cdot R, c_1 = const. \end{cases}, \delta_{sp}^{(n,k)}(t) \in [0,1]. \tag{9}$$

The degree of temporal correlation is defined in a similar way, considering two consecutive states of a specific node *n* for two timestamps *t* and $t+\Delta t$. The angle between the two speed vectors is $\varphi^{(n)}(t,t+\Delta t)=\angle\left(\mathbf{v}^{(n)}(t);\mathcal{M}_{\Delta t}\left\{\mathbf{v}^{(n)}(t)\right\}\right)$.

$$\delta_{tmp}^{(n)}(t,t+\Delta t)=\begin{cases}\cos\left(\varphi^{(n)}(t,t+\Delta t)\right)\dfrac{\min}{\max}\left\{\left\|\mathbf{v}^{(n)}(t)\right\|;\left\|\mathcal{M}_{\Delta t}\left\{\mathbf{v}^{(n)}(t)\right\}\right\|\right\}\\ 0,\text{ for }\left|\Delta t\right|>c_2, c_2=const.\end{cases}, \tag{10}$$

where $\delta_{tmp}^{(n)}(t,t+\Delta t)\in[0,1]$. The constants $c_{1,2}$ are used to make the map the discrete characteristics of the model. By applying the two degrees of spatial and temporal correlation for a specific node *n* and its history of neighborhoods, the following selection parameter is obtained:

$$\Pi_4^{(n)}(t)=\underbrace{\frac{\sum\limits_{\forall k\in\mathcal{V}^{(n)}(t)}\delta_{sp}^{(n,k)}(t)}{\left|\left\{(n,k)\middle|\delta_{sp}^{(n,k)}(t)\neq 0,\forall k\in\mathcal{V}^{(n)}(t)\right\}\right|}}_{spatial}+\underbrace{\frac{\sum\limits_{\Delta t=-N+1}^{0}\delta_{tmp}^{(n)}(t,t+\Delta t)}{\left|\left\{\Delta t\middle|\delta_{tmp}^{(n)}(t,t+\Delta t)\neq 0,\forall\Delta t\in\overline{-N+1...0}\right\}\right|}}_{temporal} \tag{11}$$

- **Node average relative speed –** this parameter is based on the relative speed between node *n* and a specific neighbor *k*, $\mathbf{v}_r^{(n,k)}(t)=\left\|\mathbf{v}^{(n)}(t)-\mathbf{v}^{(k)}(t)\right\|$; a weighing factor is applied to ensure that $\Pi_5^{(n)}(t)\in[0,1]$; the negative value is considered to align the maximization of all six selection parameters, considering that relative speed should be minimized.

$$\Pi_5^{(n)}(t)=-\frac{\sum\limits_{\forall k\in\mathcal{V}^{(n)}(t)}\mathbf{v}_r^{(n,k)}(t)}{\left|\left\{(n,k)\middle|\mathbf{v}_r^{(n,k)}(t)\neq 0,\forall k\in\mathcal{V}^{(n)}(t)\right\}\right|}. \tag{12}$$

- **Average connectivity duration** – it is an alternative link quality parameter based on the estimated duration of connectivity between two neighboring nodes *n* and *k*, $dc^{(n,k)}(t)$, given by (1). There are two consecutive levels of averaging: for all nodes in current neighborhood $\forall k\in\mathcal{V}^{(n)}(t)$ and for all previous stable neighborhoods stored in the history $\forall\mathcal{V}_{st}^{(n)}(\tau)\in\mathcal{H}^{(n)}(t)$.

$$dc^{(n)}(t) = \frac{\sum_{\forall k \in \mathcal{V}^{(n)}(t)} dc^{(n,k)}(t)}{\left|\left\{(n,k) \middle| dc^{(n,k)}(t) \neq 0,\right\}\right|}, \Pi_6^{(n)}(t) = \frac{\sum_{\tau=t-N+1}^{t} dc^{(n)}(\tau)}{\left|\left\{\tau \middle| dc^{(n)}(\tau) \neq 0, \forall \tau \in \overline{t-N+1...t}\right\}\right|}. \quad (13)$$

The metric is designed to be **maximized**, in order to obtain the "best" cluster-head selection. Together with the OSCAR implementation, the metric can be tuned to accommodate other selection parameters. Weights can be setup adaptively, depending on the environment. This adaptation is controlled by the Cross-Layer logic in the proposed NGVN architecture. The following section describes the main threads of the OSCAR algorithm.

5. OSCAR algorithm implementation

The OSCAR algorithm based on the new defined selection metric $SM^{(n)}(t)$ is built around three main processing streams, implemented as dedicated threads in NS-3 network simulator. In order to exchange mobility information between nodes, beacons are sent periodically. Unlike the DBC approach in [3] where 2 types of beacons are sent (one for topology discovery and one for nodes partitioning), in OSCAR implementation the network overhead is reduced by using a single type of beacon sent. Considering T_S the beaconing interval, an additional update interval is defined $T_{CHU} = \beta \cdot T_S$, where $\beta \in \mathbb{N}^*$, $\beta > 1$. Within a full interval T_{CHU}, $\beta - 1$ discovery messages will be sent, together with one update message (for partitioning). For simplicity and without introducing additional constraints, we consider that $\beta \in \mathbb{N}^*$. The main threads are described in the tables below.

Table 1

Main threads for OSCAR algorithm

main ***NGVN_OSCAR_LOM*** ()
{ **CLCM→setup** T_S, T_{CHU}, T_{sim}, T_c, $w_i(t)$, R, TTL, $H_{\max} = \left|\mathcal{H}^{(\bullet)}(t)\right|$, λ;
CLCM→setup G_{thr}, Π_1^{thr}, nM_{thr}, δ_{SM}, δ_c;
$\gamma \leftarrow 0$; $\beta \leftarrow T_{CHU} / T_S$;
Thread #1 : SendBeaconBundle ();
Thread #2 : ReceiveBeaconBundle ();
Thread #3 : CleanUpNeighbourhood (); }

The first thread **SendBeaconBundle()** sends beacons for neighborhood discovery, as well as update messages for clusters partitioning. The logic is described in Table 2.

Table 2

Procedure to send beacons for neighborhood discovery and cluster update

Thread #1 : *void* ***SendBeaconBundle*** $(n, t, T_S, T_{CHU}, w_i(t))$
{ **while** $t \le t + T_{sim}$ { $\gamma \leftarrow \gamma + 1$;
if $\gamma < \beta$ { $bUpd \leftarrow 0$; $tip^{(n)}(t) \leftarrow null$; $da^{(n)}(t) \leftarrow null$; }
else
{ $bUpd \leftarrow 1$; $\mathcal{V}_{st}^{(n)}(t) \leftarrow \mathcal{V}^{(n)}(t)\Big\vert_{lq^{(n)}(t) \ge lq_{thr},\, dc^{(n,k)}(t) \ge dc_{thr}}$;
$G^{(n)}(t) \leftarrow NC_{st}^{(n)}(t) = \left\vert \mathcal{V}_{st}^{(n)}(t) \right\vert$; **update** $\Pi_i^{(n)}(t)$, $i = \overline{1,6}$ **in** { (6) ... (13) };
compute selection metric $SM^{(n)}(t)$;
if $G^{(n)}(t) > G_{thr}$ && $\Pi_1^{(n)}(t) > \Pi_1^{thr}$ { $da^{(n)}(t) \leftarrow da^{(n)}(t) + 1$;
if $\left\vert \mathcal{H}^{(n)}(t) \right\vert == H_{\max}$ { $\mathcal{H}^{(n)}(t) \leftarrow \mathcal{H}^{(n)}(t) \setminus \left\{ \mathcal{V}_{st}^{(n)}(\tau_{\min}) \right\}$;
$\mathcal{H}^{(n)}(t) \leftarrow \mathcal{H}^{(n)}(t) \cup \left\{ \mathcal{V}_{st}^{(n)}(t) \right\}$; } } **else** { $da^{(n)}(t) \leftarrow 0$; }
if $da^{(n)}(t) == 0$ {
$tip^{(n)}(t) \leftarrow nV$; **reset** $CH^{(n)}(t)$, $EID_{CH}^{(n)}(t), SM_{CH}^{(n)}(t)$; }
else if $da^{(n)}(t) > nM_{thr}$ { $tip^{(n)}(t) \leftarrow nM$; }
else { $tip^{(n)}(t) \leftarrow nC$; } $\gamma \leftarrow 0$; }
send beacon [n, $EID(n)$, t, $SM^{(n)}(t)$, $TTL = 1$, R, $\mathbf{p}^{(n)}(t)$, $\mathbf{v}^{(n)}(t)$, $\boldsymbol{\theta}^{(n)}(t)$, $CH^{(n)}(t)$, $EID_{CH}^{(n)}(t), SM_{CH}^{(n)}(t)$, $dtn(n)$, $repo(n)$, $ste^{(n)}(t)$, $tip^{(n)}(t)$, $da^{(n)}(t)$, $bUpd$]; **wait** T_S; $t \leftarrow t + T_S$; } }

The second thread **ReceiveBeaconBundle()** described in Table 3 runs for every node receiving beacons. After receiving the beacons, the information regarding the source node is updated and new parameters are computed. In this stage, the selection metric is assessed and cluster-head information is updated to the node with the highest metric.

Table 3

Procedure to receive beacons for neighborhood update and clusters update

Thread #2 : *void* ***ReceiveBeaconBundle*** (k, t, **from** n, T_S)
{ **while** $t \le t + T_{sim}$ {
if $bUpd == 0$ {

$ts^{(k)}(n) \leftarrow t$;
if $n \in \mathcal{V}^{(k)}(t)$ {
$nb^{(k)}(n) \leftarrow nb^{(k)}(n)+1$; **exponential smoothing for** $\mathbf{v}^{(n)}(t)$, $lq^{(n)}(t)$;
update $(n; EID(n); \mathbf{p}^{(n)}(t); \mathbf{v}^{(n)}(t); \boldsymbol{\theta}^{(n)}(t); ts^{(k)}(n); nb^{(k)}(n); lq^{(k)}(n))$; }
else { $nb^{(k)}(n) \leftarrow 1$; $NC^{(k)}(t) \leftarrow NC^{(k)}(t)+1$;
add $\{(n; EID(n); \mathbf{p}^{(n)}(t); \mathbf{v}^{(n)}(t); \boldsymbol{\theta}^{(n)}(t); ts^{(k)}(n); nb^{(k)}(n); lq^{(k)}(n))\}$ to $\mathcal{V}^{(k)}(t)$;}
$\Delta\mathbf{p}(n,k) = \mathbf{p}^{(k)}(t) - \mathbf{p}^{(n)}(t)$; $\Delta\mathbf{v}(n,k) = \mathbf{v}^{(k)}(t) - \mathbf{v}^{(n)}(t)$;
get $dc^{(n,k)}(t)$ **using** (1); **update** $dc^{(n,k)}(t)$ **in** $\mathcal{V}^{(k)}(t)$; }
else { **if** $n \in \mathcal{V}^{(k)}(t)$ **then**
update $SM^{(n)}(t)$, $tip^{(n)}(t)$, $CH^{(n)}(t)$, $EID_{CH}^{(n)}(t), SM_{CH}^{(n)}(t)$ **in** $\mathcal{V}^{(k)}(t)$; $SM_{\max}(t) \leftarrow \max\{SM^{(i)}(t) \mid \forall i \in \mathcal{V}^{(k)}(t)\}$;
$i_{\max}(t) \leftarrow i \in \mathcal{V}^{(k)}(t)\big|_{SM^{(i)}(t)=SM_{\max}(t)}$;
if $SM_{\max}(t) > SM_{CH}^{(k)}(t) + \delta_{SM}$ {
$CH^{(k)}(t) \leftarrow i_{\max}(t)$; $EID_{CH}^{(k)}(t) \leftarrow EID^{(i_{\max})}(t)$;
$SM_{CH}^{(k)}(t) \leftarrow SM_{\max}(t)$; } } }
wait T_S; $t \leftarrow t + T_S$; } }

The third thread is the "link failure" or clean-up thread (Table 4), **CleanUpNeighbourhood()** implementing the deletion of links from the neighborhood, which have become invalid. A link becomes invalid if the nodes move outside the coverage range and no more beacons are received for a specific period of time. The implementation is based on the timestamp $ts^{(n)}(k)$ that contains the moment when the last beacon was received.

Table 4

Procedure to clean-up dead links from node neighborhood

Thread #3 : *void* ***CleanUpNeighbourhood*** (k, t, T_c)
{ **while** $t \le t + T_{sim}$ {
for $\forall j \in \mathcal{V}^{(k)}(t)$ {
if $ts^{(k)}(j) < t - \delta_c$ { $\mathcal{V}^{(k)}(t) \leftarrow \mathcal{V}^{(k)}(t) \setminus \{j\}$
if $CH^{(n)}(t) == j$ {
$SM_{\max}(t) \leftarrow \max\{SM^{(i)}(t) \mid \forall i \in \mathcal{V}^{(k)}(t)\}$; $i_{\max}(t) \leftarrow i \in \mathcal{V}^{(k)}(t)\big|_{SM^{(i)}(t)=SM_{\max}(t)}$;

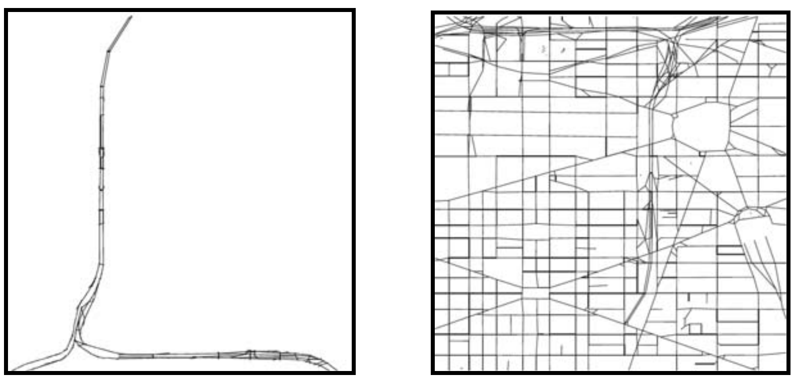
if $SM_{\max}(t) > SM_{CH}^{(k)}(t) + \delta_{SM}$ {
$CH^{(k)}(t) \leftarrow i_{\max}(t)$; $EID_{CH}^{(k)}(t) \leftarrow EID^{(i_{\max})}(t)$; $SM_{CH}^{(k)}(t) \leftarrow SM_{\max}(t)$;
} } } } **wait** T_c; $t \leftarrow t + T_c$; } }

The three threads listed above concentrate the main logic of the OSCAR algorithm. The flexibility of the algorithm resides in the large number of parameters which can be used to tune the behavior: periods for discovery beaconing, update beaconing, simulation and neighbor clean-up - T_S, T_{CHU}, T_{sim}, T_c; selection parameter weights controlling the metric - $w_i(t)$; connectivity radius – *R*, transmission power and beacons lifetime *TTL*; cyclic buffer size for history of neighborhoods - $H_{\max}$; weight for exponential smoothing - λ; link quality and connectivity duration thresholds - lq_{thr} and dc_{thr}; threshold for number of stable links per node - G_{thr}; similarity threshold between current node neighborhood and history of stable node neighborhoods - Π_1^{thr}; threshold for "Member" typology setup - nM_{thr}; selection metric thresholds - δ_{SM}; clean-up duration threshold - δ_c.

6. Simulations and results

In order to assess the performance of OSCAR, DBC algorithm is used as benchmark. This is due to the fact that DBC outperforms other clustering algorithms like Lowest-id, weighted or non-weighted Highest-connectivity (degree), Least Cluster Change (LCC) and Distributed Mobility-Adaptive Clustering (DMAC). Results shown below will prove the superior performance of OSCAR in comparison with DBC, therefore in comparison with the other relevant clustering algorithms listed above. The most important performance criteria used in the analysis are: average cluster size - $\overline{\dim}_{cl}$ and average number of clusters - $\overline{n}_{cl}$; cluster membership degree - $\overline{p}_{\%CN}$ and average rate of cluster-head changes per node- $\overline{R}_{CH,change}$; average cluster membership lifetime per node - $\overline{D}_{nM}$ and average cluster lifetime - $\overline{D}_{CH}$; overhead per node - $\overline{OH}_n$ and number of typology status changes - $\overline{n}_{st.ch.}$ In the following, n_{total} is the node density within the simulation areas.

In order to generate the simulation models and implement the OSCAR algorithm, a combination between NS-3 as network simulator and VanetMobiSim (Vehicular Ad Hoc Networks Mobility Simulator) as mobility traces generator was used. VanetMobSim traces constitute input data for NS-3. For the spatial model, two types of topologies were used: one for highway scenarios, one for metropolitan scenarios, both described in Fig. 1 a) and b). The two types of topologies were imported from TIGER/files, provided freely by the U.S. Census Bureau. Scenario S1 is a highway portion of roughly 10km, with 2+1 lanes per driving direction. Each node is equipped with an 802.11 device, with a transmission range depending on the emitting power and propagation model (150m to 300m). The number of vehicles ranges from 250 and 1000 and the speed between 80km/h and 130km/h (22 m/s and 36m/s). The mobility pattern is an overlap between RWP (Random Waypoint) model, IDM (Intelligent Driver Model) and MOBIL models. Simulation interval was set to 1800s and average speed to 100km/h. Scenario S2 describes the metropolitan topology, using an urban surface of 3km x 3km including several roads and intersections with traffic lights. Speed limits were set depending on the type of road, between 50 km/h and 70 km/h. Transmission range varied between 150 and 300m and for propagation three models: Two-Ray Ground, Rayleigh and Rice.



a) b)

Fig. 1. Simulation areas : a) S1 - Highway test topology ; b) S2 - Metropolitan test topology

In the first set of simulations performed, the maximum range was varied by changing the transmit power. The purpose was to verify the impact in cluster stability, cluster numbers and sizes. The weights of the selection metric $w_i(t)$ were considered all to be equal to 1. Discovery beacon period was set to $T_S = 2s$ and $T_{CHU} = 10s = 5T_S$. The link quality threshold was set to $lq_{thr} = 0.6$ and $\delta_{SM} = 0.1$. The storage capacities and DTN capabilities were considered constant and equal for all nodes throughout the simulations. The average rates of cluster-head change $\overline{R}_{CH,change}$ are shown in Fig. 2a and 2b. Results obtained show an increase for $\overline{R}_{CH,change}$ with the increase of n_{total}.

Fig. 2a. $\overline{R}_{CH,change}$ variation with R and n_{total} - scenario S1

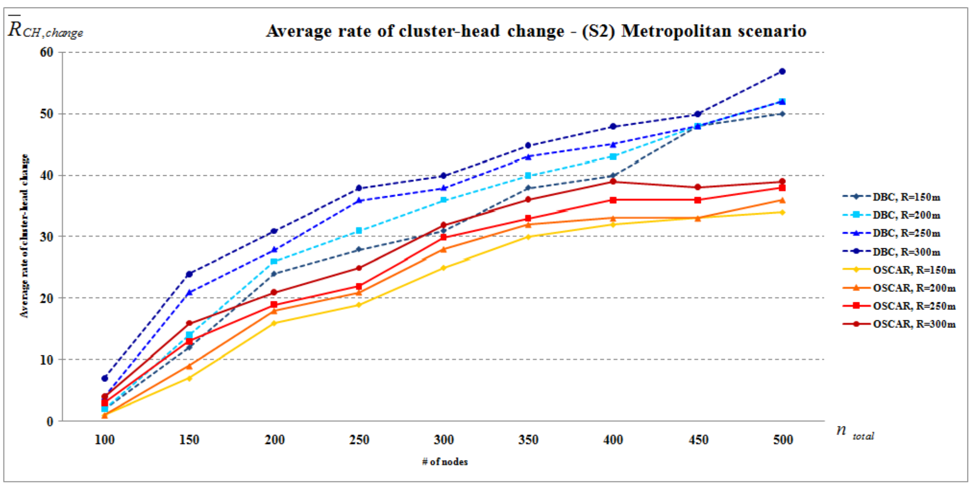


Fig. 2b. $\overline{R}_{CH,change}$ variation with R and n_{total} - scenario S2

Also, as the connectivity radius R increases, $\overline{R}_{CH,change}$ increases. This is due to the increased number of candidate nodes for cluster-head selection. The graphs show the superior performance in the case of OSCAR, for which **cluster-head variation is smaller** (more stable structures). This is due to the complex selection metric introducing greater stability by enabling multiple selection parameters. Unlike for S2, in S1, although the number of nodes is higher, the specific topology favors the decrease of $\overline{R}_{CH,change}$ and there is no linear dependency between network size and $\overline{R}_{CH,change}$. A smaller value for $\overline{R}_{CH,change}$ shows an increased cluster stability.

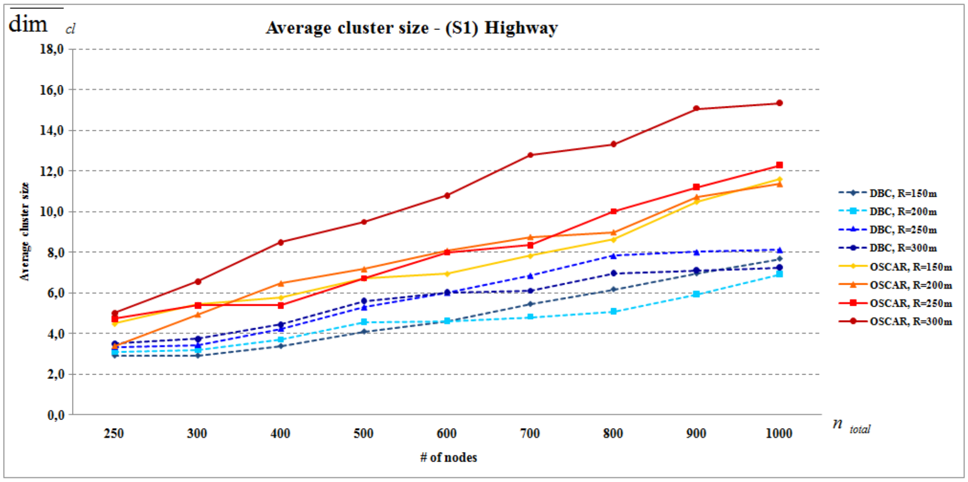


Fig. 3a. $\overline{\dim}_{cl}$ variation with R and n_{total} - scenario S1

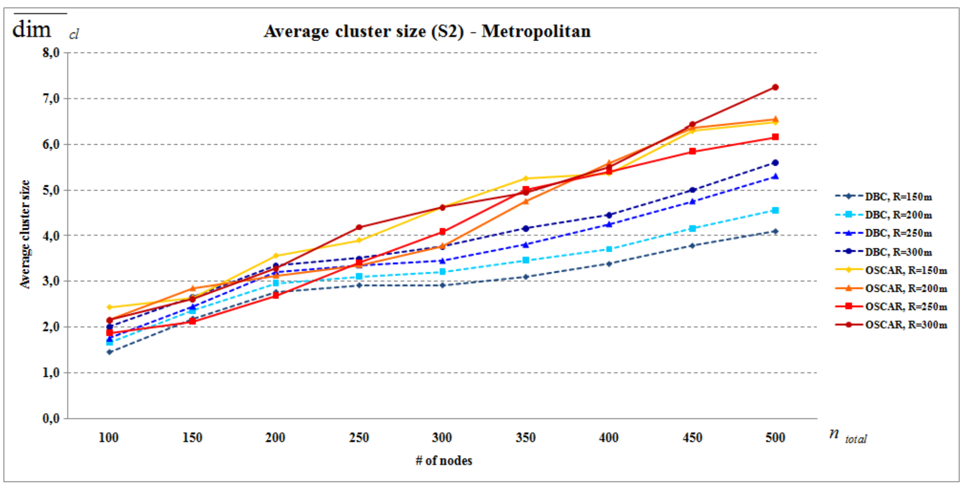


Fig. 3b. $\overline{\dim}_{cl}$ variation with R and n_{total} - scenario S2

For the same case, we analyze the behavior of the average cluster size $\overline{\dim}_{cl}$ when varying the range and the total number of nodes. Results are described in Fig. 3a and Fig. 3b. Results obtained show that as R increases, $\overline{\dim}_{cl}$ increases and OSCAR outperforms the DBC algorithm in terms of average cluster size. Maximum cluster size is obtained for maximum connectivity radius, in tradeoff with the rate of cluster-head changes which is also high. The next graphs show the correlated number of clusters $\overline{n}_{cl}$ variation (Fig. 4a and Fig. 4b).

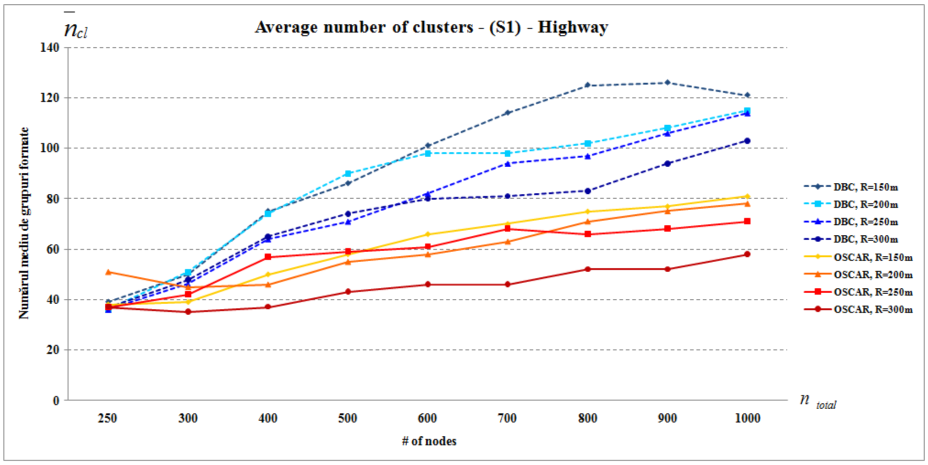


Fig. 4a. $\overline{n}_{cl}$ variation with R and n_{total} - scenario S1

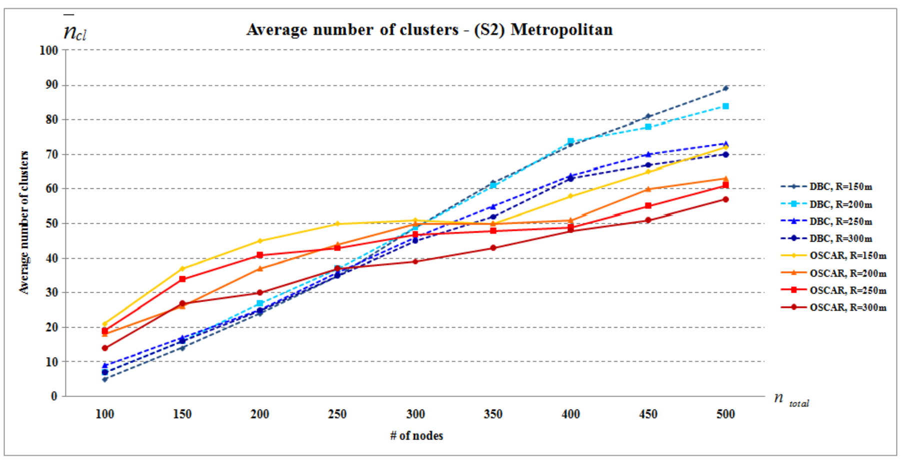


Fig. 4b. $\overline{n}_{cl}$ variation with R and n_{total} - scenario S2

The smaller the connectivity range is, the smaller $\overline{\dim}_{cl}$ is, due to more restrictive conditions for stable links, lower connectivity duration. As $\overline{\dim}_{cl}$ decreases, $\overline{n}_{cl}$ increases. $\overline{n}_{cl}$ is an indicator of algorithm performance and in the case of OSCAR, the average number of clusters is smaller, so clustering efficiency exceeds DBC.

The second set of simulations consists in a variation of the discovery beacon period T_S. This period is varied between 1s and 5s (60…12 beacons / minute) and 5s. The range is considered 250m. The rest of the parameters are kept constant. The same three metrics $\overline{R}_{CH,change}$, $\overline{\dim}_{cl}$ and $\overline{n}_{cl}$ are assessed. The variations are described in Fig. 5a, 5b and 5c for the metropolitan scenario S2.

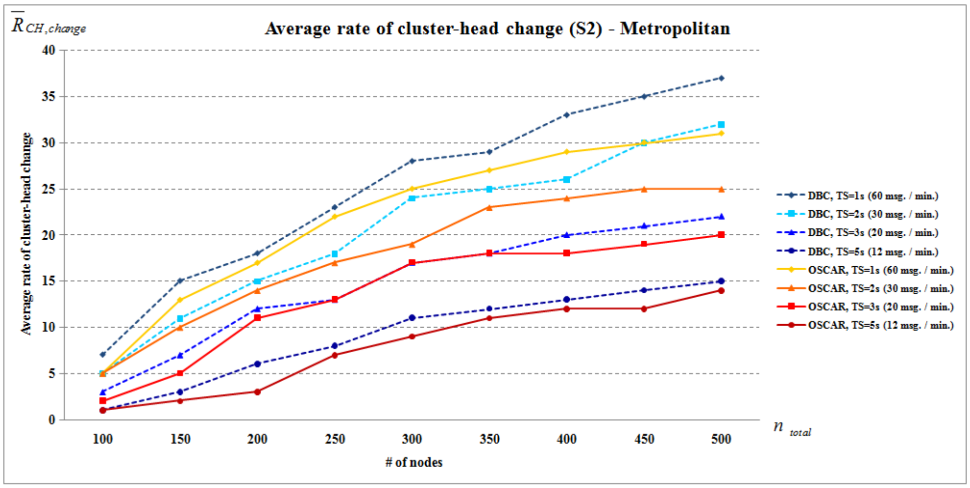


Fig. 5a. $\overline{R}_{CH,change}$ variation with T_S and n_{total} - scenario S2

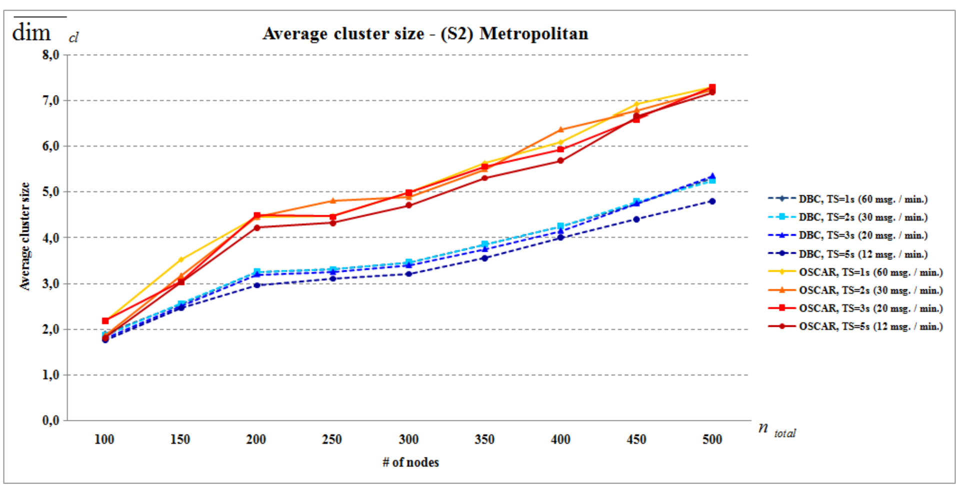


Fig. 5b. $\overline{\dim}_{cl}$ variation with T_S and n_{total} - scenario S2

As T_S increases the number of beacons decreases. When $\overline{R}_{CH,change}$ decreases, the cluster becomes more stable. The growth tendency for $\overline{R}_{CH,change}$ is less visible for higher node densities than for smaller ones. The performance of OSCAR exceeds DBC and the clusters obtained are larger. This behavior can be explained by the increased number of checks performed part of the complex selection metric, before cluster setup. Also, when n_{total} increases, $\overline{\dim}_{cl}$ increases and $\overline{n}_{cl}$ decreases.

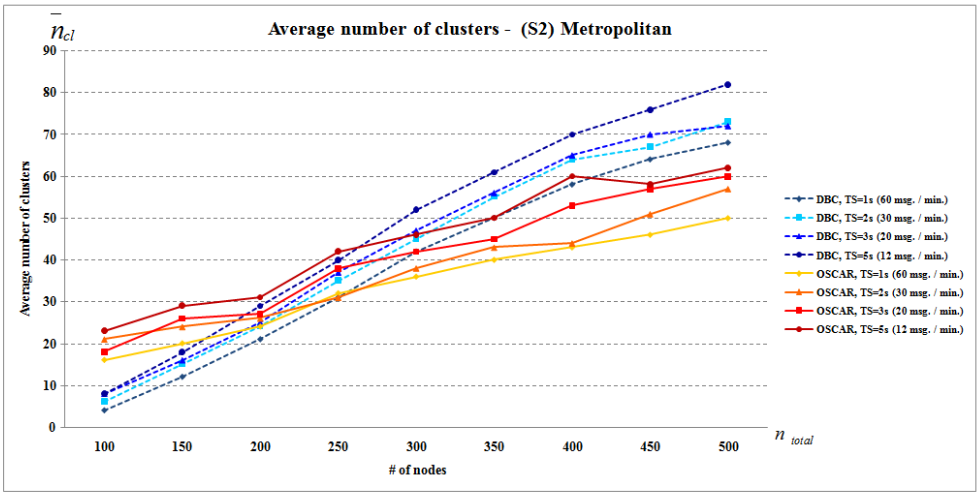


Fig. 5c. $\overline{n}_{cl}$ variation with T_S and n_{total} - scenario S2

7. Conclusions

The paper proposes a new mathematical model for the definition of a new complex optimized cluster-head selection metric and an implementation of a new clustering algorithm based on it (OSCAR), in the framework of Next Generation Vehicular Networks defined in [1] and [2]. The implementation and the simulations were performed in NS-3, considering various environment (both highway and metropolitan). The results of the ample set of simulations in NS-3 show that OSCAR algorithm outperforms the DBC algorithm proposed in [3], in terms of average cluster size (higher), rate of cluster-head changes (lower), total number of clusters (lower), translated in increased cluster stability, higher percentage of clustered nodes and larger cluster sizes.

As the clusters become more stable, the communication between cluster members is improved so typical VANET routing algorithms can be applied for intra-cluster routing (like AODV – Ad-Hoc On-Demand Distance Vector or OLSR – Optimized Link State Routing). Multipath routing algorithms are also applicable on top of the OSCAR clustering scheme. Towards the exterior of the clusters, for the overall vehicular network, the stable structures obtained can be virtualized and represented as NGVN Abstract Nodes. Such virtual nodes would inherit the mobility context information of the cluster-heads and become Custody Managers in the Disruption Tolerant forwarding nodes set, enabling complex vehicular applications. Specific DTN information dissemination mechanisms (like epidemic forwarding) can be applied in order to secure the messages delivery from source nodes to destination nodes.

Acknowledgements

This paper is the result of the scientific research activity performed by the author as part of the POSDRU 5159 program “PhD students supporting innovation and competitiveness” financed from POSDRU funds as per contract POSDRU/6/1.5/S/16.

R E F E R E N C E S

[1] *A. Matei, G. Wolny and S. Kukliński*, “A Software-Oriented Architecture for Next Generation Vehicular Networks” in Proceedings of 2010 Third International Conference on Communication Theory Reliability and Quality of Service (CTRQ2010), June 2010, pp. 37–42, doi 10.1109/CTRQ.2010.14

[2] *S. Kukliński, A. Matei and G. Wolny*, “NGVN: A framework for Next Generation Vehicular Networks”, in Proceedings of the 8th International Conference on Communications (COMM2010), June 2010, pp. 297 – 300, doi 10.1109/ICCOMM.2010.5509082

[3] *S. Kukliński and G. Wolny*, “Density based clustering algorithm for VANETs”, in Int. J. Internet Protocol Technology (IJIPT 2009), **vol. 4**, no. 3, 2009, pp. 149-157

[4] *J. Yu and P. Chong*, “A survey of clustering schemes for mobile ad hoc networks”, in IEEE Communications Surveys and Tutorials, **vol. 7**, 2005, pp. 32-48

[5] *Stefano Basagni*, ”Distributed clustering for ad hoc networks”, in Proceedings of 4th International Symposium on Parallel Architectures, Algorithms and Networks, 1999 (I-SPAN99), 1999, pp. 310-315

[6] *C. Shea, B. Hassanabadi and S. Valaee*, ”Mobility-based Clustering in VANETs using Affinity Propagation”, IEEE Global Telecommunications Conference 2009 (GLOBECOM 2009), Nov. 30 2009-Dec. 4 2009, pp. 1-6, doi 10.1109/GLOCOM.2009.5425236

[7] *M. S. Almalag and M. C. Weigle*, ”Using Traffic Flow for Cluster Formation in Vehicular Ad-hoc Networks”, Proceedings of the 2010 IEEE 35th Conference on Local Computer Networks (LCN'10), IEEE Computer Society, pp. 631-636, doi 10.1109/LCN.2010.5735785

[8] *P. Fan, J. Haran, J. Dillenburg and P. C. Nelson*, ”Traffic Model for Clustering Algorithms in Vehicular Ad-Hoc Networks”, 3rd IEEE Consumer Communications and Networking Conference 2006 (CCNC 2006), pp. 168 – 172, doi 10.1109/CCNC.2006.1593009

[9] *B. Ramakrishnan, R. S. Rajesh and R. S. Shaji*, ”CBVANET: A Cluster Based Vehicular Adhoc Network Model for Simple Highway Communication”, Int. J. Advanced Networking and Applications, **vol. 02**, issue 04, 2011, pp. 755-761