Landmarks Selection Algorithm for Virtual Coordinates Routing

Sergey Baskakov

Information Systems and Telecommunications Department
Bauman Moscow State Technical University
Moscow, Russian Federation
sergey.baskakov@mail.ru

Abstract. In this paper, we propose a distributed, self-organized landmarks selection algorithm which ensures different patterns of landmarks spread throughout deployment area of a wireless sensor network. The algorithm is highly scalable through decentralized implementation with low time and memory complexity. The proposed technique represents an optimal complexity algorithm for virtual coordinates routing protocols in large-scale wireless sensor networks, and our simulations show that it improves significantly virtual coordinates routing protocols performance, preserving simplicity and high scalability of this routing method.

1 Introduction

Routing in wireless sensor networks (WSN) is one of the most challenging and actual research areas, since routing protocol performance has a significant impact on overall network efficiency. Many emerging WSN applications and data dissemination methods (e.g. data-centric storage) require scalable and reliable point-to-point routing service. Such service could be implemented with either real or virtual coordinates routing techniques.

Geographic routing protocols, e.g. GPSR [1], use physical space location information, the real coordinates, for greedy packet forwarding. The protocols have been considered as one of the most promising solutions for providing point-to-point routing in large-scale WSNs, because they operate via only local interactions between neighboring nodes and require constant per-node state. However, geographic routing protocols efficiency degrades under realistic operation in presence of voids and localization errors.

Recently, Virtual Coordinates Routing (VCR) approach has been proposed to retain the stateless point-to-point routing ability and to eliminate the shortcomings of the traditional geographic routing technique. VCR protocols such as BVR [2], LCR [3,4], Hop ID [5], VCap [6] and HGR [7] assign each node virtual coordinates - a vector of hop counts to a small fixed set of reference nodes called landmarks (also called beacons or anchors). Virtual coordinates serve as geographic locations for greedy forwarding by minimizing the distance between the current node and the destination. In case a local minimum is reached, the protocols use a backtracking mode to guarantee packet delivery at the expense of path length increase.

Y. Li et al. (Eds.): WASA 2008, LNCS 5258, pp. 17–28, 2008.

Virtual coordinates are based on connectivity information and not on physical positions and distances; therefore, VCR protocols are insensitive to voids and location errors. Moreover, a routing protocol may automatically adapt to network topology dynamics through a periodic refresh of virtual coordinates.

Known VCR protocols vary by details such as virtual distance metric, landmarks count and their selection scheme, as well as backtracking mode algorithm. The overall protocol efficiency is predetermined primarily by greedy forwarding success rate, which depends on distance metric and landmarks distribution pattern. In this paper, we cover in detail the landmarks selection problem, a challenging issue that has been only minimally addressed by other works on VCR.

The landmarks selection problem is to find such a landmark placement over network deployment area that will result in a high success rate of greedy routing. Obviously, it is desirable to use as fewer landmarks as possible in order to minimize overhead due to storage (per-node state) and transmission (per-packet state) of virtual coordinates. Moreover, since no manual configuration is possible in most application scenarios, the self-organizing nature of WSN requires automatic landmarks selection from normal nodes.

Beacon Vector Routing [2] and Hop ID [5] protocols select landmarks randomly. Such scheme has simple implementation, but it cannot guarantee uniform landmark distribution. For this reason, we should use more landmarks (up to several dozens) to reduce the protocol's sensitivity to a randomly sampled reference nodes placement. Thus, the random landmarks selection approach yields stable greedy mode efficiency at the expense of the protocols' higher overhead.

Works such as [4,6,7] propose usage of boundary nodes as landmarks and show that such placement improves VCR protocol performance as compared to the random selection scheme.

In [6], boundary nodes are automatically found by a special algorithm. However, the algorithm is capable to select only 3 landmarks; its expansion for a larger number of landmarks requires more complicated heuristics.

The original paper on LCR [3] assumes that landmarks are manually placed at the boundaries of the deployment area. Obviously, manually controlled landmarks selection complicates large-scale WSN deployments or is even impossible in many applications scenarios. Hence, in the later work [4], authors proposed a distributed and self-organized algorithm that allows selecting any necessary number of landmarks along the network perimeter. Later we show, however, that the LCR landmarks selection algorithm's drawback is dependence of its memory complexity on the network scale.

Thus, algorithms, proposed in other works besides [4], do not provide automatic selection of any fixed number of landmark nodes in wireless mesh network. In this paper, we fill in the gap by proposing a distributed algorithm for automatic landmarks selection, taking place both at the network initialization phase and after possible landmarks failures during the operation. Depending on a chosen version, the algorithm ensures that landmarks are evenly spread throughout the deployment area or placed uniformly along the topology boundary. At the same time, our algorithm has lower memory complexity than the analogue from [4].

2 Landmarks Selection Algorithm

We consider wireless sensor network consisting of n nodes randomly placed on a plane covering A m^2 area. We assume that network topology is static and nodes have a radio communication range of r m. Thus, the average network density, i.e. the average neighborhood size, is $\rho = n\pi r^2/A$ nodes. We estimate the network diameter d - the maximum value among lengths of the shortest paths between all nodes pairs - with the following formula,

$$d \approx \frac{\sqrt{A}}{r} = \sqrt{\frac{\pi n}{\rho}}. \tag{1}$$

Note, if nodes are deployed in 3-dimensional space with volume A_{3D} m^3, the network diameter is given by

$$d \approx \frac{\sqrt[3]{A_{3D}}}{r} = \sqrt[3]{\frac{4}{3}\pi\frac{n}{\rho}}, \tag{2}$$

but anyway the proposed algorithm does not require any revisions.

The landmarks selection algorithm chooses n_L ($n_L \leq n$) landmark nodes from original nodes set V ($|V| = n$), resulting in landmark nodes set V_L ($V_L \subseteq V$). The landmarks selection criterion depends on their desired distribution. Current algorithm versions may produce one of two landmark placement patterns:

– even spread out at maximum distances from each other;
– uniform placement along the network boundary.

At first, we describe briefly the centralized algorithm implementation to show its general idea. Then we give details for its distributed implementation.

2.1 Centralized Algorithm Implementation

General algorithm structure could be divided into 3 sequential phases.

Initiator Node Selection. At this stage, we choose randomly one node that initiates a landmarks selection procedure by broadcasting beacon packets, receiving such packets all other nodes calculate their distances to the initiator node. In this paper, we define distance between the nodes as the shortest path length in a hops count; however, other link metrics are admissible without any need for algorithm modification.

The First Landmark Selection. After all the nodes have measured distances to the initiator, the most distant node is selected as the first landmark, i.e.

$$l_1 = \operatorname*{argmax}_{v \in V} h_0(v), \tag{3}$$

where $h_0(v)$ - distance between node v and the initiator node.

The selected node starts to function as the landmark, and the initiator becomes an ordinary node after receiving the first beacon packet from the landmark l_1.

Other Landmarks Selection. Subsequent landmarks selection is based on a special voting function, which value for a node defines its priority to be declared as a landmark.

The voting function may have different appearances depending on the desired landmark placement. In this paper, we offer two voting functions:

$$f_{min}(\boldsymbol{v}, L) = \min_{i \in L} v_i \tag{4}$$

and

$$f_{prod}(\boldsymbol{v}, L) = \prod_{i \in L} v_i, \tag{5}$$

where
$\boldsymbol{v}$ - virtual coordinates vector of node $v \in V$, $\boldsymbol{v} = \{v_i\}_{i=1}^{n_L}$;
v_i - i-th coordinate of node v;
L - set of active landmarks indices.

The second and next landmarks are selected sequentially: the k-th landmark is the node with the maximum votes, i.e.

$$l_k = \operatorname*{argmax}_{v \in V} f(\boldsymbol{v}, L_k), 2 \leq k \leq n_L. \tag{6}$$

This iterative algorithm results in a landmarks set $V_L = \{l_1, l_2, \dots, l_{n_L}\}$. If we use voting function (4) in (6), landmarks will be evenly spread throughout the deployment area, and function (5) selects landmarks among nodes located at the network boundary.

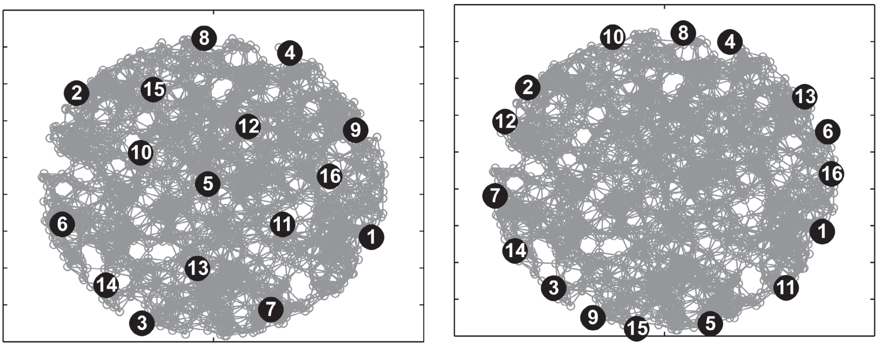
Figures 1 and 2 demonstrate landmarks selection algorithm output for large-scale dense and small-scale sparse networks. Landmarks are denoted as large circles with indicated assignment sequence numbers.

Placing landmarks at maximum distances from each other prevents their concentration in separate network areas, so both voting functions tend primarily to select landmarks from boundary nodes as these nodes are the most mutually distant ones. Then, however, $f_{min}(v, L)$ function exploits nodes in the network center, such strategy results in uniform landmarks spread throughout the deployment area. On the other hand, voting function $f_{prod}(v, L)$ selects all landmarks from boundary nodes, which causes even placement of the landmarks along the network perimeter.

Voting functions use only those vector components that correspond to landmarks active at the moment of votes calculation. Thus, in case of the sequential selection, active landmarks indices set is equal to $L_k = \{1, 2, \dots, k-1\}$.

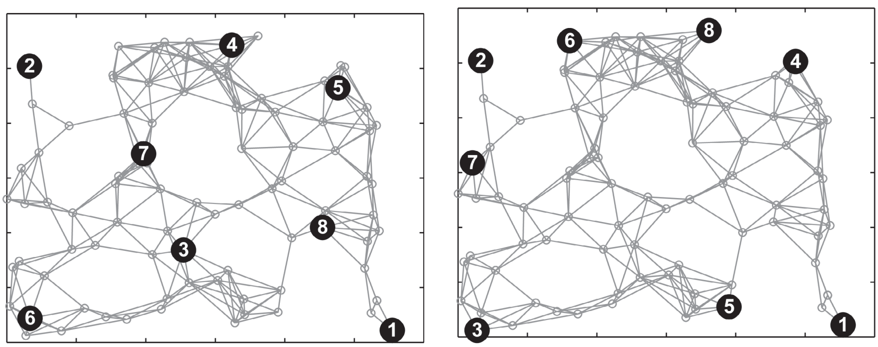
Obviously, it is possible that several nodes have equal maximum number of votes. Such conflict may be resolved by various techniques. For instance, considering uniqueness of nodes identifiers, or addresses, a node with a higher, or lower, address gets priority.

Landmark Substitution. One or more landmarks may fail during network operation. In such cases, substitution for the m-th landmark can be found with (6) calculated over set $L = \{1, 2, \dots, m-1, m+1, \dots, n_L\}$.



(a) With voting function $f_{min}(v, L)$ (b) With voting function $f_{prod}(v, L)$

Fig. 1. Landmarks selection results for large-scale dense network



(a) With voting function $f_{min}(v, L)$ (b) With voting function $f_{prod}(v, L)$

Fig. 2. Landmarks selection results for small-scale sparse network

2.2 Distributed Algorithm Implementation

The distributed algorithm implementation is intended for practical use and implies that each node performs the same set of operations without any centralized control.

Initialization. At startup, each node $v \in V$ initializes its virtual coordinates $\boldsymbol{v} = \{v_i = \infty\}_{i=1}^{n_L}$, clears the neighborhood table and sets timers for two independent periodical processes:

- local neighborhood exchange process with period t_n;
- landmarks selection process with period t_{ls}.

At this moment, all nodes are equal, there are no landmarks in the network and nodes should select among themselves n_L landmarks in a distributed manner.

Local Neighborhood Exchange Process. In Local Neighborhood Exchange process nodes periodically broadcast special beacon packets to discover their neighbors and to update connectivity information according to the actual topology state. Beacon packets broadcasting is used for virtual coordinate system construction like in any VCR protocol, and we just put into such packets some additional information described further. Local neighborhood exchange period t_n defines reactivity to topology dynamics, and thus it can be considered as the VCR protocol parameter independent of the network scale.

Landmarks Selection Process. This process means that each node checks with period t_{ls} the necessity to vest itself with a function of landmark. Every time the timer expires each node $v \in V$ executes the following algorithm.

If v is already the landmark (or initiator), or all n_L landmarks are elected, it stops current iteration and resets the timer.

If v does not know any landmarks, i.e. $L(v) = \emptyset$, it declares itself as the initiator node and also exits this procedure.

If the initiator node or some landmarks ($|L(v)| < n_L$) are already active, then the index of the next selected node is

$$k = \min\left\{i : 1 \leq i \leq n_L, i \notin L(v)\right\} \tag{7}$$

and node v calculates its votes to become the k-th landmark:

$$p\left(\boldsymbol{v}, L(v)\right) = \begin{cases} h_0(v), & \text{for } k = 1; \\ f\left(\boldsymbol{v}, L(v)\right), & \text{for } 1 < k \leq n_L. \end{cases} \tag{8}$$

Similarly, node v takes count of votes for its neighbors from the subset $\{w : w \in N(v), L(w) = L(v)\}$ (where $N(v)$ is the total set of v's one-hop neighbors) and finds neighbor q with maximum votes count $p\left(\boldsymbol{q}, L(q)\right)$. If inequality

$$p\left(\boldsymbol{v}, L(v)\right) > p\left(\boldsymbol{q}, L(q)\right) \tag{9}$$

holds, v declares itself as the k-th landmark. If v and q have an equal number of votes, we also can use their addresses for conflict resolution as described above.

In its beacon packets, the landmark includes a votes count and set $L(v)$ that took place at the moment of its election. We denote these values as $\hat{p}_i$ and $\hat{L}_i$ for the i-th landmark.

Obviously, the described landmarks selection procedure could be executed both periodically and asynchronously on detection of landmark failure, but t_{ls} value should be such that all nodes have enough time to compute distance to the recently selected landmark. The minimum permissible value of t_{ls} is equal to the worst case delay of beacon packet propagation from a landmark to all other nodes.

Landmarks Priorities Rules. A situation where several landmarks present with equal indices is obviously unavoidable, because nodes analyze only local information while making a decision about the landmarks selection. Therefore, we propose the landmarks prioritizing algorithm to suppress redundant landmarks in a distributed manner.

As described above, nodes periodically broadcast beacon packets, and at reception of such packet from neighbor q node v gets the following information about q:

- virtual coordinates vector $\boldsymbol{q} = \{q_i\}_{i=1}^{n_L}$;
- set of known to q landmarks indices $L(q)$;
- for each i-th landmark $l_i(q)$ known to q, i.e $i \in L(q)$:
  - number of votes $\hat{p}_i(q)$ at the moment of $l_i(q)$ appointment;
  - known landmarks indices set $\hat{L}_i(q)$ at the moment of $l_i(q)$ appointment.

After reception of beacon packet from q, node v executes the following procedure for each i-th landmark.

If v and $l_i(q)$ are both i-th landmark, but v had fewer votes, i.e. $\hat{p}(v) < \hat{p}_i(q)$, it stops functioning as a landmark and becomes ordinary node.

If ordinary, i.e. non-landmark, node v has no information about i-th landmark, it accepts node $l_i(q)$ as a landmark only in case of a lower votes count, i.e. the inequality $p(v, \hat{L}_i(q)) < \hat{p}_i(q)$ is checked. Notice that we use $\hat{L}_i(q)$ set to calculate v's votes for such comparison.

If v already knows another i-th landmark, it agrees to accept $l_i(q)$ as the new one only if $l_i(q)$ had more votes than $l_i(v)$ at the moment of appointment, i.e. in case the inequality $\hat{p}_i(v) < \hat{p}_i(q)$ holds.

3 Algorithm Complexity Analysis

In this section, we estimate time and memory complexity of the distributed algorithm implementation. Time complexity stands for the total time required to select n_L landmarks in a newly deployed network. Memory complexity is the amount of each node's memory resources required to maintain the algorithm state.

We assume that the initiator node selection overhead is negligible. The first and the following landmarks are selected in sequence with time steps proportional to the worst case delay of beacon packet propagation $O(dt_n)$, but the distributed algorithm implementation requires additional time reserve to suppress duplicate landmarks. Therefore, taking into account that t_n is the routing protocol setting parameter independent of network scale, we get an equation for the algorithm time complexity:

$$T = O\left(2n_L d\right). \tag{10}$$

The (10) is the convergence time of the distributed algorithm and relates to any network topology and arbitrary (not only uniform) nodes placement over deployment area.

On average, each node has ρ one-hop neighbors and should maintain information about them in order to decide about declaring itself as the landmark. Thus, the algorithm memory complexity is equal to

$$M = O\left(n_L \rho\right), \tag{11}$$

because per-neighbor state overhead is $O\left(n_L\right)$.

In general, VCR protocols set n_L as a constant independent of total nodes count n, and in the majority of cases the relation $n_L \ll n$ holds. Therefore, we have the following algorithm complexity estimates for large-scale networks:

$$\tilde{T} = O\left(d\right), \tag{12}$$

$$\tilde{M} = O\left(\rho\right). \tag{13}$$

Obviously, any VCR protocol (regardless of landmarks assignment method) requires $O\left(d\right)$ time to construct a virtual coordinates system, because such time is necessary for beacon packets to propagate throughout the network, and $O\left(\rho\right)$ memory to store the neighborhood table. Hence, if some distributed landmarks selection algorithm has $O\left(d\right)$ execution time, uses $O\left(\rho\right)$ memory to maintain data about one-hop neighbors and results in both landmarks selected and virtual coordinates calculated by every node, then such algorithm is *optimal* for application in VCR protocols.

Therefore, the proposed landmarks selection algorithm is optimal for VCR protocols in large-scale networks, in which $n_L \ll n$ and n_L is a constant.

4 Comparison with LCR Algorithm

Cao et al. proposed originally Logical Coordinate Routing protocol in [3] and they complemented it with distributed landmarks selection algorithm in [4]. Their algorithm also allows to assign any fixed number of landmarks from boundary nodes but uses different voting function and selection procedure.

The LCR landmarks selection algorithm and the proposed here one, when used with voting function $f_{prod}(v, L)$ (5), induce almost similar placement of landmarks, but differ in time and memory complexity. Therefore, we compare only algorithmic complexity of these two landmarks selection techniques. Unfortunately, there is no algorithm complexity analysis in [4], so we have performed the analysis ourselves and here present only the final results for large-scale networks, omitting the detailed algorithm description for brevity purposes.

Assuming the inequality $n_L \ll n$ holds and n_L is a constant, the LCR landmarks selection algorithm time and memory complexity are equal to

$$\tilde{T}_{LCR} = O\left(d\right), \tag{14}$$

$$\tilde{M}_{LCR} = O(\rho + n/\rho) = O(\rho + d^2) \tag{15}$$

for 2D-space placement and

$$\tilde{M}_{LCR} = O(\rho + n/\rho) = O(\rho + d^3) \tag{16}$$

for 3D-space deployment.

Time complexity of both algorithms is proportional to network diameter d, because $O(d)$ is the worst-case propagation delay of a beacon packet; however, our algorithm executes in $2n_L$ times longer (see (10)) than the LCR landmarks selection algorithm as we select landmarks in series. However, in case of the large-scale networks, its time complexity could also be estimated as $O(d)$. Thus, both algorithms are optimal for VCR protocols in terms of time complexity.

In terms of memory complexity our algorithm is absolutely scalable as its memory requirements depend only on nodes deployment density ρ and are not affected by total nodes count n or network diameter d. Algorithm exploits only local information about one-hop neighborhood that is also required by any VCR protocol, so in a sense of memory complexity, it is optimal for virtual coordinates routing.

On the other hand, according to (15) and (16) the LCR landmarks selection algorithm memory complexity depends linearly on network size n and quadratically (or in third-degree for 3D-space deployment) on diameter. Therefore, it does not have the scalability property to the full extent, and its application in large-scale networks may entail implementation difficulties.

5 Simulations

We implemented basic virtual coordinates routing framework and distributed version of the proposed landmarks selection algorithm in discrete event simulation system, OMNeT++ ver. 3.2 combined with Mobility Framework ver. 2.0, to study expedience of algorithm introduction into VCR protocols. We simulate performance of only a network layer, not taking into account such issues as packet losses due to errors, collisions, buffers overflows, etc.

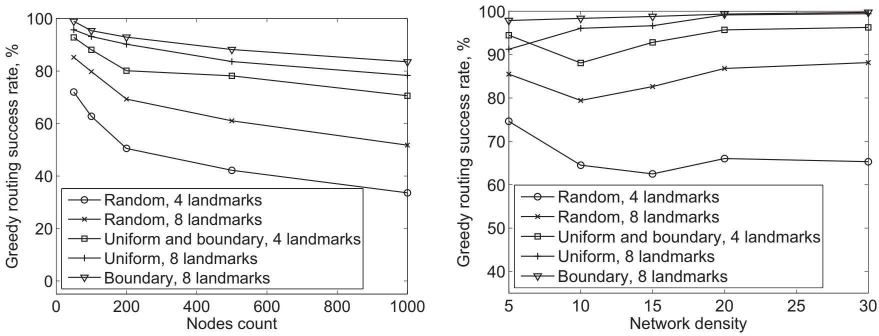
In our simulations, we generate random network topologies according to given parameters (nodes count n, network density ρ and diameter d). Nodes are uniformly distributed over the deployment square field with area A m^2 and communication range $r = 50$ m. We randomly choose 100 nodes to be source and destination pairs, so the results of each simulation run are averaged over 9900 paths. If network size is less than 100 nodes, all of them exchange packets. Ordinary Euclidean norm is used as a virtual distance metric.

We compare effectiveness of virtual coordinates routing technique under 3 different landmarks placement strategies:

- random placement;
- uniform spread throughout deployment area (Fig. 1(a));
- network boundary placement (Fig. 1(b)).

Figure 3(a) shows greedy mode packets delivery success rate under variable nodes count n under fixed network density $\rho = 10$ nodes (network diameter varies from 8 to 35 hops). Figure 3(b) demonstrates impact of one-hop neighborhood size ρ under fixed network diameter $d = 10$ hops (nodes count varies from 23 to 358).

The number of landmarks n_L is set to 4 and 8. If landmarks count is 4, the algorithm gives almost the same landmarks placement for both voting functions (see Fig. 1), therefore, results, obtained for uniform and boundary placement of 4 landmarks, are shown in common plots.



(a) Variable nodes count at fixed density (b) Variable density at fixed diameter

Fig. 3. Packets delivery success rate as a function of network parameters

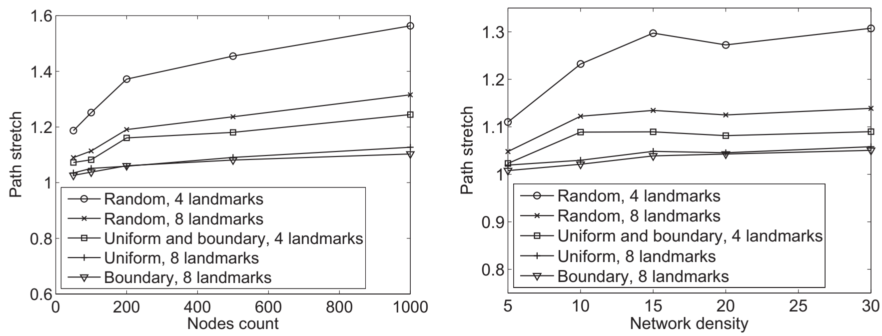
For all strategies of landmarks distribution success rate decreases monotonically as the network size grows (Fig. 3(a)), but at uniform or boundary placement the slope is smaller than at random landmarks selection. The reason for such a success rate decrease is that network extension under fixed density causes growth of network diameter and average path length. As a result, there arises a probability of a local minimum occurrence during packet delivery process.

On the other hand, as network density increases, the routing protocol performance improves, saturating at $\rho > 20$ nodes (Fig. 3(b)). If landmarks are spread out uniformly through the deployment area or placed along the network boundary, the greedy routing success rate reaches values of more than 95% at 4 landmarks and above 99% at $n_L = 8$, whereas for random landmarks selection these values are 66% and 87% respectively.

The second VCR protocol performance metric is path stretch - the ratio of found routing path length to the shortest path length. Although it is known that the shortest path is not always optimal under unreliable and asymmetric links, we use here hop count metric because of accepted ideal PHY and MAC layers assumption. If greedy forwarding fails due to the local minimum, we use backtracking mode from [2]. It is significant to mention that delivery success rate is 100% under all settings provided that backtracking mode is enabled.

The results, presented in Fig. 4, demonstrate direct relationship between greedy mode success rate and routing paths length. Low success rate means a high probability of a local minimum, hence routing protocol switches to backtracking mode more frequently resulting in paths stretching. At 4 landmarks placed according to the algorithm, paths are 9% to 25% shorter than paths under random landmarks selection, whereas at 8 landmarks the advantage is slightly lower, 4% to 19%.

Thus, the introduction of the proposed landmarks selection algorithm into VCR protocols improves significantly efficiency of greedy forwarding over virtual coordinates. At the same time, greedy routing success rate dispersion is much lower than in random landmarks placement; therefore, VCR protocol performance will be more stable and predictable.



(a) Variable nodes count at fixed density (b) Variable density at fixed diameter

Fig. 4. Path stretch as a function of network parameters

Our simulations also reveal that landmarks placement over the network boundary produces better results than uniform spread throughout the deployment area.

6 Conclusion

We proposed a distributed and self-organized algorithm intended for automatic selection of any fixed number of landmarks both at network initialization phase and after possible landmarks failures during operation. Described voting functions provide two different landmarks distribution strategies, even spread throughout deployment area and uniform network boundary placement. Other voting functions could also be introduced into the algorithm framework to obtain alternative desired landmarks placement patterns.

The algorithm complexity analysis showed that it is both time and memory complexity optimal for large-scale WSNs, whereas the memory complexity of its analogue from [4] depends considerably on the network size.

We demonstrated through simulations that introduction of the proposed landmarks selection algorithm into virtual coordinates routing protocol improves efficiency of greedy forwarding, preserving simplicity and high scalability of this routing technique.

We suppose that the proposed algorithm could be also useful in other WSN research areas (for example, improving accuracy of distributed localization methods), but it is a subject of future work.

References

1. Karp, B., Kung, H.: GPSR: greedy perimeter stateless routing for wireless networks. In: Proceedings of the 6th annual ACM/IEEE international conference on mobile computing and networking, Boston, Massachusetts, USA, pp. 243–254 (2000)
2. Fonseca, R., Ratnasamy, S., Zhao, J., Ee, C., Culler, D., Shenker, S., Stoica, I.: Beacon vector routing: scalable point-to-point routing in wireless sensornets. In: Proceedings of the 2nd symposium on networked systems design and implementation, Boston, Massachusetts, USA (2005)

3. Cao, Q., Abdelzaher, T.: A scalable logical coordinates framework for routing in wireless sensor networks. In: Proceedings of the 25th IEEE international real-time systems symposium, pp. 349–358 (2004)
4. Cao, Q., Abdelzaher, T.: Scalable logical coordinates framework for routing in wireless sensor networks. ACM Transactions on Sensor Networks 2(4), 557–593 (2006)
5. Zhao, Y., Li, B., Zhang, Q., Chen, Y., Zhu, W.: Efficient hop ID based routing for sparse ad hoc networks. In: Proceedings of the 13th IEEE international conference on network protocols (2005)
6. Caruso, A., Chessa, S., De, S., Urpi, A.: GPS free coordinate assignment and routing in wireless sensor networks. In: Proceedings of the 24th annual joint conference of the IEEE computer and communications societies, vol. 1, pp. 150–160 (2005)
7. Liu, K., Abu-Ghazaleh, N.: Aligned virtual coordinates for greedy routing in WSNs. In: Proceedings of the 3rd IEEE international conference on mobile ad-hoc and sensor networks, Vancouver, BC, Canada, pp. 377–386 (2006)