

# GENETIC GENERATION OF INTERNET OF THINGS OVERLAY NETWORK INFRASTRUCTURE

Rares MANIU<sup>1</sup>

<sup>1</sup>Eng., Ph.D. (c), Military Technical Academy, 39-49 George Cosbuc Bvd., Bucharest, Romania

**Abstract:** *The Internet of Things now represents one of the biggest technology wave in information and communication technology domain. In such a network, the communication parameters are critical and, for improving its potential, the utilization of an overlay network infrastructure is the clue. This paper proposes a genetic generation of such a network topology, based on existent parameters of communication infrastructure and taking in consideration the quality of services demands*

## INTRODUCTION

Because of evolution in data acquisition, processing and communication, a new concept named Internet of Things (IoT) achieve attention.

Due to the interaction between people and smart things and, because the communication capabilities of devices with computational power, the Internet of Things concept tends to become a normal reality in almost all domains of life (industry, home, health care, logistics, smart cities, etc.). From this perspective, IoT is a collection of interconnected devices, uniquely addressable based on communication protocols. This means that there are a huge number of different devices that, without human involvement, interact each other, to realize a particular purpose. Because of the heterogeneity of devices, in terms of data acquisition, computing and communications, the use of them in a large information system involve a lot of problems. These problems are related to the interoperability of devices, to the need to ensure the independent functioning and to provide a certain degree of adaptability. It also should also be considered as an efficient use of resources, which imposes several constraints (operation with reduced energy reserves, having reduced capacity computing, with communication networks often made ad hoc or realized over existing networks).

In terms of autonomy, of data acquisition and interoperability, the types of sensors, of other devices used and the scope are crucial.

In terms of the communication requirements, in general, IoT applications involve interaction between user and a physical object and data transfer over a communications structure. It can be a public network (Internet), a dedicated network or a dual layer network.

If some public networks used, exists the possibility that there may be moments when it cannot provide the required communication parameters. Moreover, due to the high degree of heterogeneity of computing elements is possible that they cannot function optimally within a public network.

The option to develop dedicated networks for IoT applications is less feasible in terms of development and maintenance costs, especially when it comes to a wide area.

Considering these arguments, a dual layer network (an overlay network) based on a public network seems to be a good choice for IoT applications. In fact, an overlay network (Fig.1.) is a virtual network of nodes and logical links that are placed on top of an existing network with the purpose to implement a network service that is not available in the existing network or to modify the parameters of an existing network.

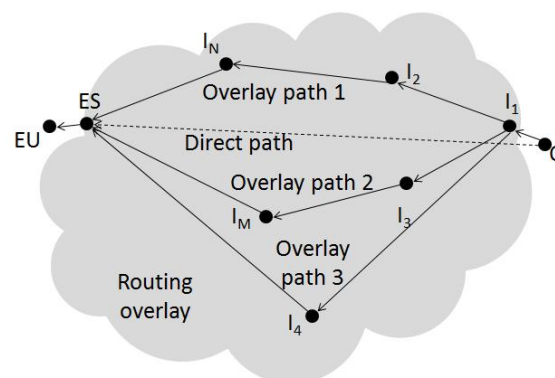


Fig.1. Architecture of an overlay network [8]

As benefits, an overlay network can improve the parameters of a public network such as reliability and delay. We do not need new devices, do not need to be deployed on every node and do not need to modify software or protocols.

Using an overlay network, as drawbacks, it adds overhead (ex IP over IP), adds complexity, and it may be too heavyweight for some nodes to support it. Such a network is difficult to scale.

This paper will study the possibility to generate an overlay network configuration over an existing network, using genetic techniques.

## RELATED WORK

Because IoT is now a new technology and have a general applicability, its particularities have been studied in several studies.

In [1] the authors present IoT technologies, potential applications, challenges and the future area of interests. The conclusions are that IoT will be implemented on high scale in next future, the technologies necessary to achieve the ubiquitous networks are expected to become mature and a high number of objects will be addressable and will be connected to IP-based networks.

Today's challenges will be that different networks who use different standards coexists and the current IT domain has no experiences in developing systems with huge number of objects connected to IP networks.

The problem of ensuring a communication infrastructure for IoT applications has been addressed in several studies. Taking into account the particularities of such a system, using a dual layer network seems to be the best option.

In [2] is approached Overlay Network Node placement Problem. That study demonstrates that this is a NP-Hard problem and to solve it is proposed an algorithm derived from k-median problem algorithm that was developed by Anya in [3]. This new algorithm is compared with TAG Algorithm proposed in [4] (who select overlay nodes from a set, based on a greedy strategy).

In [5] authors examine the impact of placing intermediate nodes within a network to optimize performance (in terms of time, hardware, network connectivity, etc.). The placement problem was studied in two scenarios: a scenario where a single overlay node is chosen in overlay routing (SOSR) and a scenario where multiple overlay nodes are places in overlay routing (SLOT). Thus, through optimal placement of overlay nodes, the authors have shown that the reliability can be improved through redundancy and it could increase bandwidth. In fact the experiments reveal that an adequate node placement outperform a random placement up to 100% for reliability in case of SOSR and up to 200% of TCP performance for SLOT.

### PROBLEM ANALYSIS AND PROPOSED SOLUTION

As in [6] and [7], the architecture of IoT can be divided in 3 levels:

- Sensors/actuators level
- Communication level
- Application level.

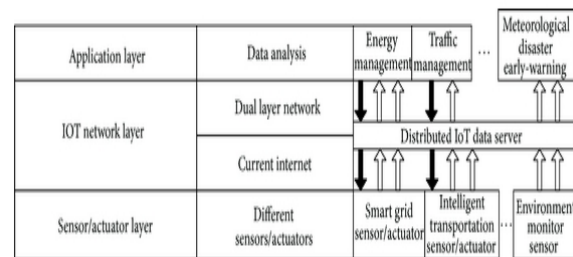


Fig.2. Architecture of IoT [2]

As previously discussed, a dual layer network based on Internet infrastructure is a solution for IoT applications. In such a network there are a number of relay nodes, placed in such a way to improve performance.

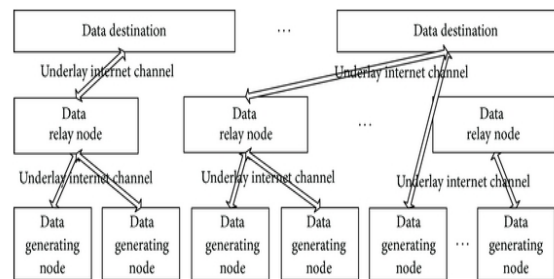


Fig.3. A dual-layer network example [2]

In mathematical terms, the placement of overlay nodes can be described using graph theory. In this way, if the network is described as a graph  $G = (N, L)$ , where  $N$  are nodes in network and  $L$  links between nodes. Each link has a cost  $C$  that is determined by link parameters. A set of  $O$  that is a subset for  $G$  contains  $M$  nodes that are potential overlay nodes. The overlay placement problem is to determine a subset of  $O$  consisting of a number of vertices that optimizes overlay routing objective function.

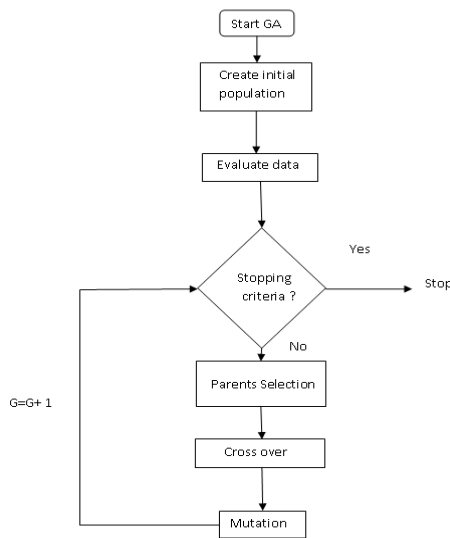
The main idea is to obtain a set of routes with minimum total costs, using a number of intermediate nodes. We take into consideration the total cost of all links from overlay network, because it assure the redundancy so, in any moment can be chosen each route. In this way, is important that all routes have minimum cost. So, if this cost drops, system performance may become better.

As operation, the overlay construction provides multiple alternative paths between source and destination. In addition, always exists the option to use the direct path, without using overlay nodes. The choice is based on best paths parameters.

Because the placement of overlay nodes is a NP hard problem, its approach is possible using genetic techniques. They can solve problems with

multiple solutions, generating a set of Pareto optimal solutions and it is efficient for optimizations. It can generate good results in multi-objective, non-linear optimizations, with non-linear restrictions, for a large seeking space, with a high number of variables.

So, using a genetic algorithm, generates a set of solutions, both direct routes and routes with overlay nodes, and, for IoT route, the route with minimum cost is chosen from this set of solutions. The function of a genetic algorithm is presented in Figure 3.



**Fig.3. A genetic algorithm steps**

In this paper will be used a generic genetic algorithm. The chromosome defines the route. Its genes are intermediate nodes between source and destination. The initial population is randomly generated.

Fitness function is an important part of a genetic algorithm and measure the quality of chromosome from a population. In this implementation the fitness function has a lower value when the fitness value of chromosome is better. It is the sum of the costs of links from o route.

About selection operator, two chromosomes are evaluated taking in consideration the value for fitness function. The better chromosome, that is the most adequate, is selected and is promoted to be in next generation.

In this paper, it is used a crossover in one-cut point. This point is chosen to be the middle node in the route. After interchange partial route of two parents, because is possible that loops are formatting during crossover, the offspring follows a procedure to eliminate them and, after, if they represent a valid route, they must be evaluated. If not, is chosen another pair of parents for

crossover, and the procedure is repeated till are obtained a full population of offspring.

The mutation operator will replace randomly, a number of chromosomes from the new population. The termination criterion is in fact the convergence of the algorithm. Because of need to evaluate the fitness evolution related to number of generations, in this implementation, the termination criterion is the reach a fixed number of generations (100).

**RESULTS**

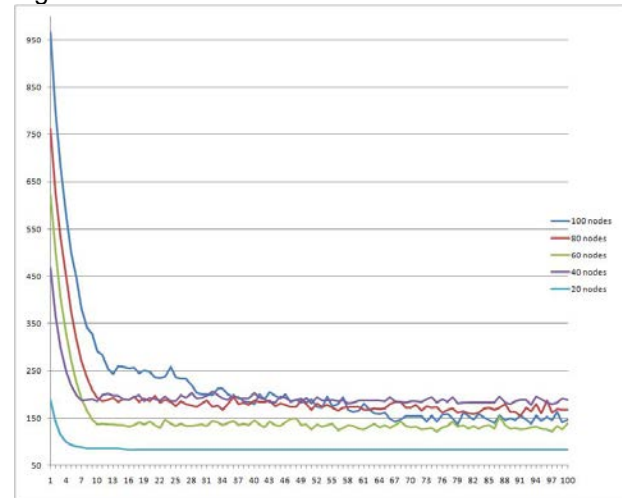
In order to test generation genetic of the overlay networks were built randomly five networks with the following parameters:

- 100 nodes, 4000 links between nodes, the links costs between 0 and 10;
- 80 nodes, 28000 links between nodes, the links costs between 0 and 10;
- 60 nodes, 1500 links between nodes, the links costs between 0 and 10;
- 40 nodes, 650 links between nodes, the links costs between 0 and 10;
- 20 nodes, 150 links between nodes, the links costs between 0 and 10;

About genetic algorithm, the generation of parents, offspring and next generation was defined with 40, 30, 25, 20 and 10 chromosomes. The maximum percent ratio for mutation was 5%. The termination criterion was number of generations (100).

Each algorithm was run 10 times on the same data set.

The evolutions of total fitness value are presented in figure nr.4. For each network, the configuration values represented in next figure are the average of fitness values obtained in ten runs of the algorithm.



**Fig.4. Total fitness evolution**

Regarding the number of routes and the number of nodes from them, in ten runs of the genetic algorithm were obtained the following results:

- About the network with 100 nodes were generated 23 separate routes with a total of 34 nodes;
- About the network with 80 nodes were generated 14 separate routes with a total of 27 nodes;
- About the network with 60 nodes were generated 12 separate routes with a total of 25 nodes;
- About the network with 40 nodes were generated x separate routes with a total of 11 nodes;
- About the network with 20 nodes were generated 6 separate routes with a total of 7 nodes.

## CONCLUSIONS

The IoT is about billions of users and is continuously developing. Processing and real-time communications for a huge number of elements will lead to the need to develop mechanisms to compensate the disadvantages of current networks.

For placing smart things on a large geographic area and also taking into account their mobility is necessary to use a large network, such Internet network to connect them. The need to respect communication parameters involves developing of an overlay network.

This large number of network elements, their dynamicity and heterogeneity, involve management tools, different from the classic ones, which provide configurations that meet the requirements in a timely manner. Genetic techniques can be used to solve this problem.

The overlay construction algorithm generates the overlay paths between source and destination by solving a multicriteria problem on an appropriately defined flow network.

The demands of each link source-destination are estimated by using a set of parameters that depends on traffic type. According with these, every link will have a cost (good links will have a lower costs). This cost is implemented in the genetic algorithm fitness function, where the important parameters will have a higher rank.

As it can be seen in results, to use a genetic algorithm is an option to generate an overlay network configuration. In Fig. 4 it can be seen that, after 40 generation, a generic genetic algorithm can present a set of good configurations (with a low and relative stable values for fitness function).

If the network is more complex (with a big number of nodes and links), the number of solutions offered by the genetic algorithm is higher. From the set of possible solutions, will be choose the most appropriate one depending on the parameters nodes (links, latencies, the possibility to implement the required software component to work as an intermediate overlay node, etc.) and application requirements.

The use of a genetic algorithm, by its intrinsic feature to explore in parallel the entire search space, offer the possibility to choose in minimum time a new configuration if topology was change (considering that in a large network, radical changes of topology are rare.).

Also, the genetic algorithm architecture provides the ability to be easily scaled if the network size changes.

## FUTURE DEVELOPMENTS

This paper examines the possibility to use genetic techniques to generate an overlay network configuration for IoT applications. The study results were obtained using generic a genetic algorithm. The future directions of the study, is the necessity to accelerate convergence of the genetic algorithm and improving the quality of results. This technique provides good results, with respect of requirements, but not necessarily the best.

The improvement can be achieved by optimizing the genetic operators (adaptive genetic operators) or by combining search with the classic genetic.

Since genetic algorithms can be parallelized by nature, a new direction of study would be to implement the algorithm on nodes, with an adequate computing capacity.

## BIBLIOGRAPHY

- [1] D. Bandyopadhyay, J. Sen, "Internet of Things - Applications and Challenges in Technology and Standardization", *Wireless Personal Communications: An International Journal*, Volume 58 Issue 1, May 2011, Pages 49-69
- [2] Y. Wan, J. Cao, K. He, H. Zhang, P. Yu, S. Yao, K. Li, "Node Placement Analysis for Overlay Networks in IoT Applications", *International Journal of Distributed Sensor Networks*, Volume 2014 (2014), Article ID 427496
- [3] V. Arya, N. Garg, R. Khandekar, A. Meyerson, K. Munagala, V. Pandit, "Local search heuristic for k median and facility location problems", *STOC'01*, July 6-8, 2001, Hersonissos, Crete, Greece
- [4] S. Roy, H. Pucha; Z. Zhang; Y. C. Hu, L. Qiu "On the Placement of Infrastructure Overlay

Nodes", IEEE/ACM Transactions on Networking, Volume 17, Issue: 4, Pages: 1298 – 1311

[5] S. Roy, H. Pucha, Z. Zhang, Y. Hu, and L. Qiu, "Overlay Node Placement: Analysis, Algorithms and Impact on Applications," in Proc. of the 27th International Conference on Distributed Computing Systems, Toronto, Canada, June 2007

[6] Valdivieso Caraguay, Ángel Leonardo, et al. "SDN: Evolution and Opportunities in the Development IoT Applications." International Journal of Distributed Sensor Networks 2014 (2014).

[7] "Internet of Things –Architecture, Final architectural reference model for the IoT v3.0", <http://www.iota.eu/public/public-documents/d1.5/view>

[8] R. K. Sitaraman, M. Kasbekar, W. Lichtenstein, M. Jain "Overlay Networks: An Akamai Perspective" <https://www.akamai.com/cn/zh/multimedia/documents/technical-publication/overlay-networks-an-akamai-perspective-technical-publication.pdf>