Journal of Theoretical and Applied Computer Science
ISSN 2299-2634 (printed), 2300-5653 (online)
Vol. 8, No. 1, 2014, pp. 3–14
http://www.jtacs.org

Game-based communication in Network Control Systems

Michal Morawski

Institute of Information Technology, Lodz University of Technology, Poland

michal.morawski@p.lodz.pl

Abstract: *Network based Control Systems (NCSs) are more and more often selected in designing distributed control systems due to both economic and practical reasons. Today designs of NCSs frequently involve the non-expensive wireless communication instead of traditional wired links. Such systems are usually called Wireless Sensor Networks (WSNs) and are used for many other purposes as well. While convenient in installation and management, wireless links are susceptible to noise and not very reliable. While common approach of data delivery relies on routing (proactive or reactive), this paper presents a different approach to designing wireless NCSs. In the proposed approach every node takes an independent decision as a result of a game between the nodes. Unlike the routing solutions, the nodes never create any path, and even have no knowledge concerning network topology.*

Keywords: *Networked Control Systems, Sensor Networks, Game Theory*

1. Introduction

The term Internet of Things (IoT) is often used as a synonym for future of communication. While it always concerns communication between some devices, IoT covers plenty of different meanings. Most often it is used in the context of creating "intelligent" environments – buildings, houses, communication, energy distribution (smart grids), etc. Internally such "intelligence" consists of many control systems connected via different kind of networks, but it is externally visible as a set of IP-based devices.

Even though in industry plants, distributed control systems are extensively used for many years, closing control loops by non-dedicated networks is discussed only recently [1]. In such systems, when actuators and sensors are placed at longer physical distances from each other or the control center, then robustness of the overall system relies on the quality of links. Well known telecommunication links (optical fibres, backhauls, etc.) provide extremely high reliability, they are expensive both in CAPEX and OPEX (i.e. according to investment and maintenance costs). Such links have high bandwidth that is crucial in typical Internet communication but is essentially needless in most of control systems. Such systems require rather low latencies, low jitter and low dropout ratio. Therefore, the classical solutions are based on various wired standards jointly called fieldbusses, which are frequently installed in real plants.

However, as the prices of wireless solutions rapidly decrease and hardware maturity increases, such solutions are more and more often considered in new implementations [2]. While wireless (most often radio but also acoustic or optical) systems show many advantages like over installation, and possibility of implementation in mobile appliances, they also have some serious limitations. In particular those limitations relate to susceptibility to noise, interference, congestion, fadings, etc., which are inevitable consequences of using shared channels. Moreover, only certain bands are accessible free of charge and Effective/Equivalent Isotropical Radiated Power (EIRP) of transceivers is always very limited due to legal restrictions, power drain limitations, etc. Therefore, these bands are typically crowded and problems with noise are straightened even though receiver sensitivity increases and required Signal-to-Noise Ratio (SNR) decreases. A non-expensive solution to this problem is application of many intermediate nodes that work as relays (forwarders) for each other.

If a relay approach is applied, then to achieve proper communication, finding a best path(s) between communicating nodes seems to be crucial. Proactive routing typically used in telecommunication networks is found undesirable due to high overhead when unreliable links are exploited and is uncommon in WSNs. On the other hand, most often applied in WSNs reactive routing works fine when only a few intermediate nodes take part in relying messages – see e.g. Routing Over Low power and Lossy networks (ROLL) [3, 4, 5]. Unfortunately, such networks show scalability limitations by design like tree structure with at most a few roots (so called sinks) and constrained number of branches and hops.

Because it is impossible to consider the network links (especially wireless ones) as reliable, whenever a message arrives at its destination, the control or supervision system must either repeat the transfer of damaged or dropped messages, or implement logic to tolerate dropouts. The first solution can be used only when large sampling periods (in comparison to network delays) can be tolerated. Otherwise, both latency and traffic intensity increase that additionally influences delays and therefore should be avoided. An example of practical (tested in a real plant) system that tolerates delays and dropouts is presented in [6]. In NCSs, constructing rely chains in control systems should be focused on reducing delays and esp. their variation (so called *jitter*) rather than on keeping low dropout ratio. Unfortunately, the application of reactive routing often violates requirement of attaining low latencies because procedure of changing the transmission path is time consuming. Note, that constant delays, even the large ones, are relatively easy to deal [7], but jitter is a serious problem.

Therefore, in this paper, we propose a new approach, different from routing, based on game between involved network nodes and other devices that use the same or different technologies but the same or overlapping communication channels. While game theory has been intensively applied to the economy problems for many years, it became recognized in network engineering only recently [8], in particular in the wireless communication area. Unfortunately, most of the proposed solutions require complex, usually floating point, computations using many readings. Such algorithms can hardly be implemented in low cost nodes with limited processing power, and therefore, we do not consider them to be practical. The solution presented in this work requires small protocol footprint, and can be implemented with low resources (CPU cycles and memory). It is worth noticing, that the presented approach is significantly different from the ones based on the behaviour of animal flocks or hives or swarms (e.g. [9]), although the behaviour of “animal” algorithms is often similar.

2. Forwarding

The idea presented in this paper is based on the one proposed by Gburzynski and Olesinski [10]. They suggested that a node should forward a packet always if it cannot find any reason to drop it and they proposed and even patented a list of “reasons”. The paper extends the concept discussed in [10] by adding different kind of reasons for packet drop and removing necessity of application most of original ones and eliminates the need for finding a route. Opposite to the original solution, the proposed one supports both unicast and multicast (including the functional multicast) communication and can be used both for mobile and stationary nodes.

In the traditional routing the paths are chosen according to some optimization procedure that (usually) minimize a metric, that is a scalar function of attributes of participating links. These attributes include hop count, reliability, energy consumption, etc. Unfortunately, unlike to guided links the wireless ones are not robust, so the routing process should be repeated very often (proactive) or precede any successful transmission (reactive). This substantially decreases performance of the network.

In this paper we do not assume transmission to follow any particular paths. After broadcasting the message by a node, this message arrives at some set (may be empty) of the remaining nodes if there is a satisfactory level of SNR during all the transmission. Frequently messages sourced by some nodes and received by common node are damaged due to interferences. The risk of such damage depends on the spatial position of involved nodes (nodes can also move), and traffic pattern of node transmissions. These information are not available at other nodes in the network. If due to any reason (node movement, congestion, etc.) the best path changes frequently path flapping/fluttering occurs – see classic papers [11, 12]. Additionally the traditional approach are prone to Pigou's and Braess'es paradoxes [13]. Generally these phenomena are recognised as a problem, but in this work we are going to turn it as an advantage.

The proposed solution joins OSI Layer 2 and Layer 3 functionalities. First of all, unlike the typical approach it was decided **not** to acknowledge the reception of a message by intermediate nodes (except the destination unicast node, that broadcasts the special "End of path" – EOP – message). Moreover, due to serious node resource constraints, and the requirement of ensuring low latencies – "hot potato" or "deflection" routing [14, 15] approach, the idea initially used for some high speed networks was incorporated. The message should be either dropped or forwarded, but not stored in the intermediate node queue. This assumption allows one to easily calculate both the upper bound of network delay and delay jitter (but not the drop ratio). Additionally, we assume no mutual coordination between the nodes. Such networks can be exploited in industrial environments for control purposes if the process control algorithm is resistant to information loss.

3. Idea of the algorithm

In the proposed solution every message contains, apart from standard source (SA) and destination addresses (DA), also the address of transmitter (TA) and the address of previous transmitter (PA), and the message ID (MID). Message originator fills the desired DA (unicast or multicast), its own address in SA, PA and TA, and generates random MID. If the medium is free (see below) the sender broadcasts the message. All these addresses may be any unique identifiers including ipv4 or ipv6 ones.

Every intermediate node can either drop, or forward the message. If the decision to forward the message is taken by the intermediate node, TA field is copied into PA, and TA field is filled with the relay node own address. SA and DA stay intact. The destination node (for unicast transmissions only) sends an acknowledgement - a message with special DA (EOP). In any case, the node stores MID in circular short buffer to avoid repeating the messages. The message is also heard by preceding node. The described mechanism, illustrated in Figure 1, is used instead of the acknowledgement-based one. It also allows for shunting useless paths.

In the situation presented in Figure 1, nodes B, E, and G retransmit the message, next

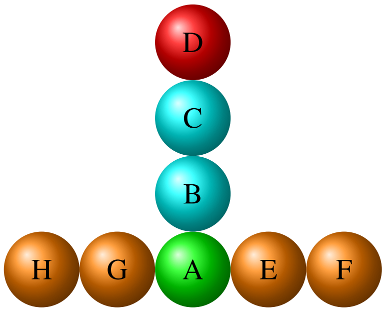


Figure 1. Node A broadcasts message with DA=D. Finally nodes B and C have to relay message, while the E, F, G and H ones should drop it. Nodes B, E and G are in range of A; none path is known by any node at any time.

nodes C, F, and H do the same. However, only node C receives the acknowledgement from node D. After some time (see section 6) both F and H will assume that there is no successor on their "paths" and will stop relaying messages (in fact, the probability of forwarding decreases), as a result, nodes E and G stop receiving acknowledgments as well and after some time stop relaying. The same mechanism can be applied also to limit multipath transmissions as presented below. Every transmission is performed only once.

4. Game

While the situation presented in Figure 1 is quite obvious, in many common situations nodes see the same message multiple times. If an intermediate (relay) node finds any reason to drop message (esp. if node is aware, that the message was already transmitted by itself or by other ones) as in [10] - the message ought to be dropped, otherwise node attempts the transmission. During this attempt, the node has to decide if it should transmit or rather drop the message because the probability to arrive to the destination is too low.

Our solution of this problem is based on the well known "Hawk and Dove" (or "chicken") game [16]. The following payoff matrix is applied (only "given node" incentives are presented, "other node" matrix is symmetrical):

		Given node	
		Forward	Drop
Other node	Forward	$-c-p$	0
	Drop	$-c+a$	$-p$

(1)

where $a, c, p \geq 0$ and $a + p > 0$. The matrix is constructed in such a way that when a message is transmitted the node always has to pay the cost of using the medium, and when the message is dropped (due to the collision when both nodes choose "Forward" or when both nodes choose "Drop") the node has to pay the penalty cost. Finally, when the message is successfully transmitted the node receives an award. The node's strategy is to maximize its profit. Coefficients a, c and p denote the award when the message arrives at the destination, the media costs, and penalty when the message is dropped on all possible paths, respectively. All theses coefficients are time-varying and are approximated by every node based on its local knowledge (without coordination with other nodes). The values of a, c and p are independent of each other. Moreover c and p are based on physical measurements (see sections 5 and 7).

Analysing payoff matrix (1), it is easy to see, that a given node should forward the message with probability q:

$$q = \begin{cases} \frac{a+p-c}{a+2p} & \text{if } a+p>c \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

Therefore the computation effort of the proposed algorithm is very low, and the algorithm can be efficiently implemented in IoT environment.

5. Media cost

We propose to evaluate the media cost c using simple low pass filter applied to continuous radio measurements when given node does not transmit the data. The media cost is given by

$$c_k = \alpha c_{k-1} + (1-\alpha)\, m_k \tag{3}$$

where k denotes subsequent time instants, m_k is the signal level indicated in receiver expressed in dBm and $\alpha \in (0,1)$ is a constant. The signal level does not depend on the modulation of transmitted frame and therefore is influenced by native communication as well as unstructured noise, or other users of the physical channel. It is simply the value of signal power on the antenna's filter pins. If the transmission is in progress the evaluation of c_k is suspended.

Because real measurements of m_k are negative (are expressed in dBm), then m_k in (3) is shifted up by the minimum sensitivity of the receiver (in the simulation studies – 88-100 dB). Theoretically, the value of m_k varies in a range near to 0 (when channel is clear) up to about 80-120 (depends on EIRP), but usable range (Δc) is narrower (we assume 35 dB).

The value of m_k changes quite fast due to environmental reasons, node mobility, unpredictable transmissions from other sources (traffic patterns, noise, etc.), or even accuracy of measurements. Therefore we propose simple autoregression filter (3), where the constant α and sampling period depend on *a priori* known environmental characteristics and application traffic pattern as well. A proper selection of the parameter value can improve the drop ratio, but this is not crucial for algorithm implementation. See simulations below, where α is intentionally set very small.

6. Award

When a message arrives at the destination, all the nodes that forwarded the message should receive the award. Unfortunately, the transmission of such information in the reverse direction is too expensive, i.e. it requires reverse acknowledgements. Therefore, we suggest to approximate the award using the following equation:

$$a_k = \begin{cases} a_{k-1} + \lambda_s & \text{if transmission was successful} \\ \max\left(a_{k-1} - \lambda_f, 0\right) & \text{if transmission was unsuccessful} \\ \max\left(a_{k-1} - \lambda_o, 0\right) & \text{if other node has forwarded} \\ \max\left(a_{k-1} - \lambda_n\left(a_{k-1} - a_{init}\right), 0\right) & \text{at clock tick} \end{cases} \tag{4}$$

where coefficients $\lambda_\star$ are positive constants. Note that the table of awards indexed by DA resembles the traditional routing table (RT). The entry in such RT is created every time a node decides to transmit the message and flushed after some time of inactivity (or exhausting RT size). $\lambda_\star$ coefficients can be interpreted as a reputation impact, where nodes behave similarly to humans in social context.

Eq. (4) is constructed taking into account the low resources (CPU and memory) of nodes, and avoiding floating point operations.

Unfortunately such approach requires manual selection of many parameters, but on the other hand a suboptimal selection is straightforward. a_{init} (initial value of the award) should be chosen based on (2) and the maximum allowable drop ratio q_{max} (that depends on the application) and possible dynamics of c.

$$a_{init} = \frac{\Delta c}{1 - q_{max}} \tag{5}$$

Simulations show, that the optimal values of $\lambda_\star$ parameters depend on traffic patterns. They should be selected to avoid unrestrained grow (i.e. stability) of a (λ_n). We recommend to sustain $\lambda_o \geq 2\lambda_f$ and $\lambda_s \approx \lambda_f$.

It is easy to notice, there are many other methods of constructing awards. For instance better (lower drop ratio) results can be obtained when increasing or decreasing the award value relative to the current award rather than absolute one (as it is observed by human behaviour) like in (4), but at the higher computation cost. We believe, the gain in achieving slightly smaller drop ratio (at most a few percent) does not justify increasing CPU consumption. Yet certainly a method of computation of the award needs further development.

7. Penalty

Not every message can be dropped at the same cost. If the message is to be dropped at its origin, the appropriate penalty is negligible. However, if multiple nodes have already forwarded the message and depleted their resources the penalty should be higher. We consider the simple hop count as inappropriate and we suggest to use the term *energy budget* that is assigned to the message by the originating node and connect it with a penalty. On the other hand the *budget* (or penalty) based on power loss only, suffers from problem depicted in Figure 2. The Figure 2 shows common situation when there are many alternative paths with similar *budgets* from node A to node D.

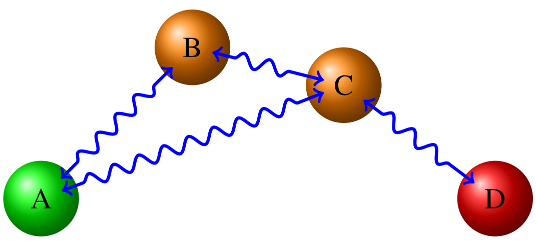


Figure 2. Node A broadcasts message with DA=D. Budget on path ABCD is approximately the same as on path ACD, which should be the preferred one.

Finally, we suggest to use penalty given by (6)

$$p = \sum_{\text{relaynodes}} (\epsilon(\text{powerloss}) + p_{hop}) \tag{6}$$

where $(\mathrm{powerloss})$ is a difference between transmit power and indicated RSSI scaled by coefficient ϵ and p_{hop} is a small value assigned by every intermediate node (e.g. assuming homogeneity of nodes and transmission power eq. 0 dBm, and RSSI of signal eq. -70 dBm, and $p_{hop} = 1$ and $\epsilon = 0.1$, relay node increases the penalty by 8).

If at any node the penalty exceeds *budget*, the message should not be forwarded. Note that (again assuming homogeneity of nodes) the difference between initial budget and penalty (current *budget*) takes the same role as TTL (Time-To-Live) in IP networks. Low current budget means that the message originator was distant, nearly (in accuracy to p_{hop}) independent on how many times the message was forwarded. Therefore, such mechanism restricts not the number of forwarders (relays) but approximates the geographical range of message validity.

8. Simulation results

In order to verify the properties of the presented solution the specialized event simulator much simpler but similar to NS2 has been developed. The simulator is of event based type and supports any pattern movement of nodes and network traffic, but is controlled by XML static configuration instead of NS2 scripts, therefore it is much faster than NS2 in considered context. In the simulator, the power loss is proportional to the square of Euclidean distance between the nodes as motivated by the simplicity and computational performance of such approach. However, other models of wave propagation have also been tested without significant impact on the results.

The results of simulation for the parallel-path topology depicted in Figure 3, are presented in Table 1. The test topology (Figure 3) is strictly symmetrical, both possible transmission paths (i.e. ABDF and ACEF) are chosen with equal probability, however, the results presented in Table 1 are limited to the situation where path ACEF wins. Analysing the game between nodes D and E, it is easy to observe that after the first unsuccessful transmissions, the probability of transmission for node D decreases significantly, and in fact that node does not take part in further communication. This situation occurs as a result of decreasing the award. On the other hand, node E transmits messages with probability close to one. After short period of time this situation is recognized by the preceding nodes. As illustrated on “Award for node B” chart, just after starting of the transmission of messages where the award increases because node B is aware of retransmissions performed by D, and then it becomes aware of the lack of retransmissions and the award diminishes. The coefficient α in media cost (3) is intentionally set not optimal for the analysed traffic pattern to show its influence on probabilities (2).

If the algorithm chooses the option “Forward”, the corresponding value of probability (2) is marked by a stem in the 3rd row, if it chooses “Drop”, the corresponding value of probability is marked in the 4th row.

Table 2 presents the results obtained using the network topology from Figure 4. For readability only the awards (equivalent of RT) with resulting forward probability are displayed. Additional traffic that disturbs the the same flow as in the previous simulation is generated (X-Y). The traffic patterns are intentionally set to maximize the number of collisions to check the algorithm robustness. The selfish behaviour of nodes C and E partially transfers traffic A-F to path ABDF to avoid overloading of links CE, AC and EF. Both flows are split over available links. This result demonstrates the load balancing property of the proposed algorithm. The table shows the most interesting case where the flow X-Y are split in ratio 55%/45% on longer path XEFDBACY. Such high (and non-intuitive) value is a result of selected traffic pattern, but

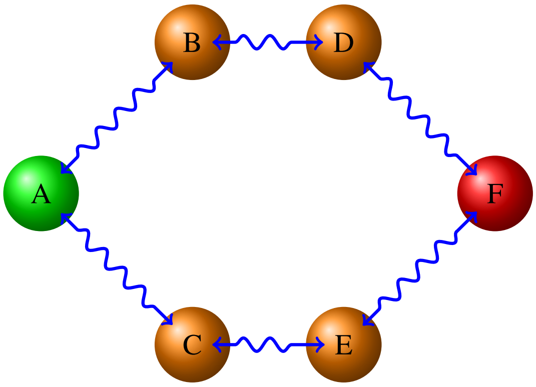


Figure 3. Node A broadcasts a message with DA=F. The message is seen only by nodes B and C. Both B and C execute the algorithm and if possible, they retransmit message. The message retransmitted by B is seen only by A and D, and the one retransmitted by C is seen by A and E only. The same situation occurs for nodes D and E. If the nodes transmit messages simultaneously, node F cannot decode the neither message. Node A starts broadcasting with uniformly distributed intervals 50 ms at 2 s and stops at 40 s.

in most cases the ratio is lower. The ratio depends on traffic pattern because of the probability of collisions.

9. Conclusion

An approach presented here is designed for low end, limited resource and inexpensive devices. It does not assume any kind of coordination between the nodes, giving extremely scalable solution for radio, acoustic and other sensor and control networks due to very efficient spectrum usage (no network control plane is required) and lack of supervision. On the other hand, packet losses are an inherent property of the algorithm, so applications on top of the network layer have to take it into account.

The algorithm presented here is **not** a routing algorithm, as a selection of a best path in the network is not performed. Although sharing certain functional similarities, it differs from routing algorithms by joining control and transport properties. Unlike a routing algorithm, it makes the forward (or drop) decision based only on the node internal information without choosing a successor. Since information about the network topology is not used for making the forwarding decisions, improved efficiency and better robustness are achieved. The presented algorithm allows for keeping the network delays in stringent bounds and provides prompt response for network events at no cost of additional bandwidth. In contrast to routing protocols, in the proposed approach drops and losses are necessary for proper work of the algorithm. It is therefore applicable only to lossy networks.

The overall performance and properties of the described solution are quite promising. The latencies are low and the drop ratio is sufficient for implementing networked control algorithms, like e.g. [6]. While the developed algorithm works correctly for a wide range of parameter values (performance only slightly decreases where the parameters are chosen outside the optimal set that changes for different traffic patterns), the way of finding the general optimal solution is left for the further research work.

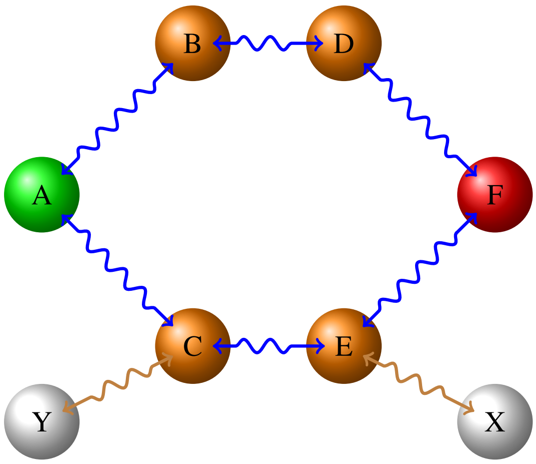


Figure 4. A topology similar to depicted on Figure 3, where nodes C and E are slightly moved to prefer path ACEF. The topology is augmented by nodes X and Y, where node X transmits data to node Y between 10th and 30th second of simulation. Background traffic is exactly the same as in previous example.

10. Acknowledgements

This work has been performed in the framework of a project *Design and validation of control algorithms in networked dynamical systems* financed by the National Science Centre of Poland – decision number DEC-2012/05/D/ST6/03030.

References

[1] Zurawski, R.: Networked Embedded Systems. Industrial Information Technology. CRC Press, New York, NY, USA, 2nd edition, 2009.

[2] Willig, A., Matheus, K., Wolisz, A.: Wireless Technology in Industrial Networks. Proceedings of IEEE, 93(6), pp. 1130–1151, 2005.

[3] Gaddour, O., Koubaa, A.: RPL in a nutshell: A survey. Computer Networks, 56(14), pp. 3163 – 3178, 2012. ISSN 1389-1286.

[4] Vasseur, J., Agarwal, N., Hui, J., Shelby, Z., Bertrand, P., Chauvenet, C.: RPL: The IP routing protocol designed for low power and lossy networks. Technical report, *http://www.cs.berkeley.edu/~jwhui/6lowpan/IPSO-WP-7.pdf*, 2011.

[5] Kilic, N., Gungor, V. C.: Analysis of low power wireless links in smart grid environments. Computer Networks, 57(5), pp. 1192 – 1203, 2013. ISSN 1389-1286.

[6] Morawski, M., Zajczkowski, A.: Approach to the design of robust networked control systems. International Journal of Applied Mathematics and Computer Science (AMCS), 20(4), p. 689698, 2010.

[7] Ignaciuk, P., Bartoszewicz, A.: Discrete-time sliding-mode congestion control in multisource communication networks with time-varying delay. Control Systems Technology, IEEE Transactions on, 19(4), pp. 852–867, 2011. ISSN 1063-6536.

[8] Han, Z., Niyato, D., Saad, W., Basar, T., Hjorungnes, A.: Game Theory in Wireless and Communication Networks: Theory, Models, and Applications. Cambridge University Press, New York, NY, USA, 1st edition, 2012. ISBN 0521196965, 9780521196963.

[9] Antoniou, P., Pitsillides, A., Blackwell, T., Engelbrecht, A., Michael, L.: Congestion control in wireless sensor networks based on bird flocking behavior. Computer Network, 57(5), pp. 1167 – 1191, 2013.

[10] Gburzynski, P., Olesinski, W.: On a practical approach to low-cost ad hoc wireless networking. Journal of Telecommunications and Information Technology, (1), pp. 29 – 42, 2008.
[11] Khanna, A., Zinky, J.: The revised ARPANET routing metric. In: SIGCOMM'89 Symposium proceedings on Communications architectures & protocols, pp. 45–56. ACM, Austin, TX, USA, 1989.
[12] Wardrop, J. G.: Some theoretical aspects of road traffic research communication networks. Proceedings of the Institution of Civil Ingineering, Part 2, 1(36), pp. 352–362, 1952.
[13] Roughgarden, T.: Selfish Routing. Ph.D. thesis, Faculty of the Graduate School of Cornell University, 2002.
[14] Feige, U., Raghavan, P.: Exact Analysis of Hot-potato Routing. In: Proceedings of the 33rd Annual Symposium on Foundations of Computer Science, SFCS '92, pp. 553–562. IEEE Computer Society, Washington, DC, USA, 1992. ISBN 0-8186-2900-2.
[15] Teixeira, R., Shaikh, A., Griffin, T., Rexford, J.: Dynamics of Hot-Potato Routing in IP Networks. In: SIGMETRICS/Performance. New York, 2004.
[16] Osborne, M.: An Introduction to Game Theory. Oxford University Press, 2000.

Table 1. Results obtained for the most important nodes in the test topology depicted on figure 3. Time is expressed in seconds, Media cost in dBm before shifting.

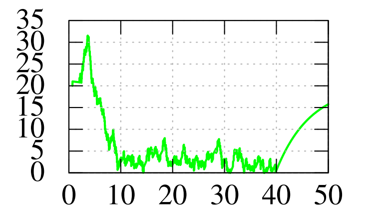
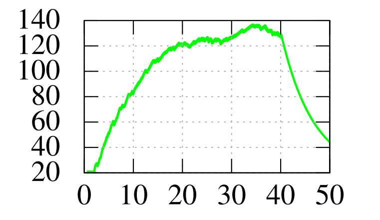
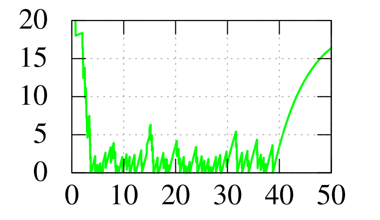
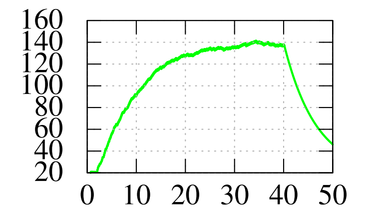
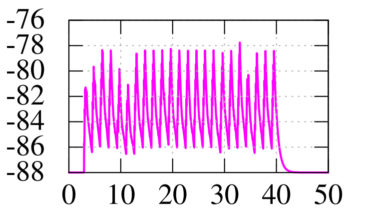
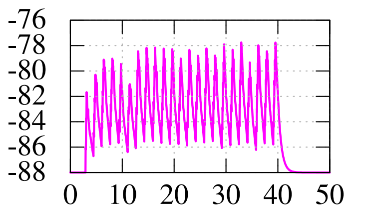
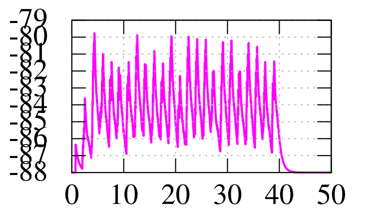
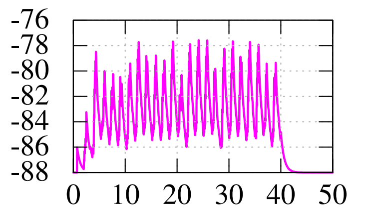
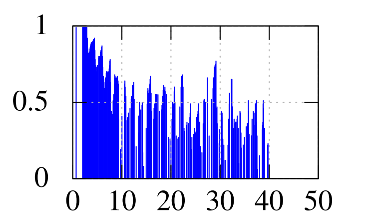
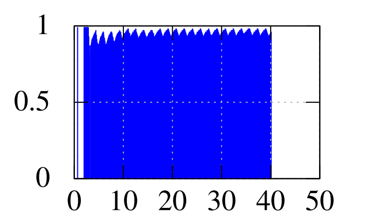
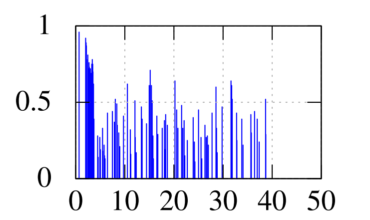
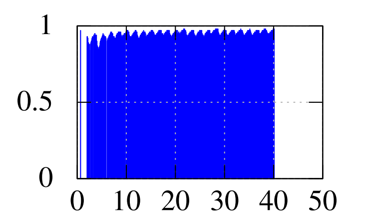
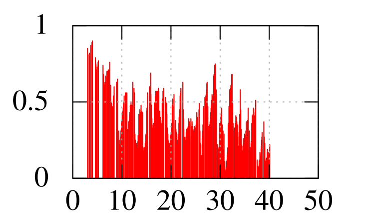
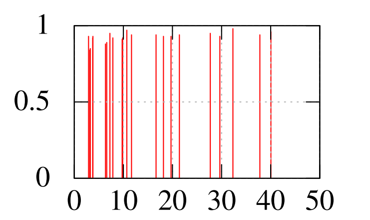
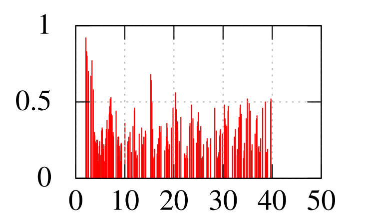
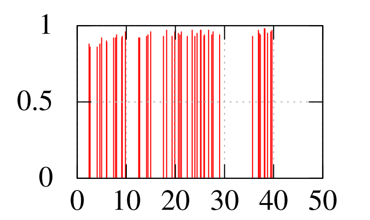
	Node B	**Node C**	**Node D**	**Node E**
Award				
Cost				
Forward probability				
Drop probability				
Drop ratio	59.9%	5.2%	66.3%	6.2%

Table 2. Results obtained for the load balancing test based on topology depicted on figure 4. Time is expressed in seconds.