Multi-agent Bio-inspired Algorithms for Wireless Sensor Network Design

Vladimir Mochalov*

* Megaputer Intelligence, Russia

sensorlife@mail.ru

***Abstract* — The process of designing a wireless sensor network (WSN) is rather complicated. This process is not formalized in the form of a hard set of rules, algorithms and standards that guarantee the construction of WSN satisfying different requirements of the designer. This paper discusses the problem of constructing a WSN structure. In the proposed functional diagram of the WSN design we can allocate the place of synthesis WSN structure functional block. Bio-inspired algorithms simulate natural processes of self-organization and evolution. The author proposes to use several multi-agent bio-inspired algorithms for synthesis of WSN structure. Fitness function performs multi-objective fuzzy expert evaluation of various WSN parameters. All considered algorithms modify the global pheromone memory. The work shows the results of the synthesis of WSN topology on an object with space constraints. The results illustrate the possibility of using different self-organization animal models to solve some problems arising in the process of building self-organizing wireless sensor networks.**

***Keywords* — sensor networks, graphs, bio-inspired algorithms, network synthesis**

I. Introduction

Distinctive features of wireless sensor networks (WSNs) include the self-organization property, the wireless data transmission environment, hardware and software limitations and stand-alone power supply of the network nodes, fixed allocation of the network nodes, focus on the problems of monitoring and remote management of distributed objects in a limited area, small amount of data transmitted by the network.

By the physical size of allocation area, the WSNs belong to a class of wireless personal area networks (WPAN). Currently, the construction of a WSN is most commonly controlled by the IEEE 802.15.4 standard. Other standards are adopted too.

The IEEE 802.15.4 standard defines two types of WSN nodes: (1) reduced-function devices (RFD) that collect information in some neighborhood of their location; (2) full-function devices (FFD) that implement both packet retransmission and information gathering. The IEEE 802.15.4 standard is a basis for higher-level protocols (ZigBee, 6loWPAN, DigiMesh, WirelessHART, etc.) and allows constructing with the help of software add-ons at the network layer and above the following topologies: point-to-point, star, cluster tree and mesh network [1].

Construction of the WSNs requires solving various complicated problems that relate to different research areas. Some of such problems are given in Fig. 1.

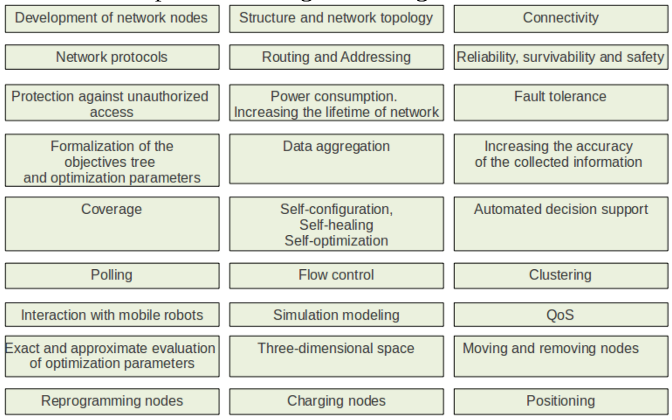

Figure 1. Several problems arising when constructing the WSN structure

II. The problem of WSN structure design in a heterogeneous space

In the work we use a model of the WSN structure, where on the functional level the following types of WSN nodes can be defined: (1) functional nodes (F-nodes) that collect information in some neighborhood of their location; (2) transit nodes (T-nodes) that manage routing and retransmit the information collected by F-nodes to the information collection centers (ICC) to be utilized further; (3) ICCs that manage the WSN and process information collected by the WSN.

A wireless sensor network is allocated on an object distributed in 2-dimensional heterogeneous space. The points of the object that must have F-nodes allocated at them are given. The heterogeneous space defines spatial restrictions on allocating the WSN nodes and the function of electromagnetic signal attenuation in this space. The use of two-dimensional space does not influence the generality of the reasoning for three-dimensional space, but simplifies theoretical consideration of the proposed method and software implementation of the algorithm for designing fault-tolerant structure of WSN.

In general case there can be multiple ICCs in the WSN, and the information that has arrived into each of them is available to one or multiple users for making decisions and performing certain actions. It means that information received by F-nodes should be retransmitted, with a required degree of reliability, to several ICCs by means of transit nodes allocated within the given object in a certain way.

We will consider that this information is used by a generalized “end user”, for instance, is being sent from each ICC to the Internet.

We know the description of the WSN allocation object, spatial restrictions for allocating the WSN nodes, a finite set of different, intercompatible types of functional nodes N_F, transit nodes $N_{T\ and}$ information collection centers N_C. Also, we know the allocation of different types of F-nodes and information collection centers (Fig. 2). It is necessary to allocate T-nodes in such way (Fig. 3), that the designed WSN structure would have the «desired properties» assigned by a designer.

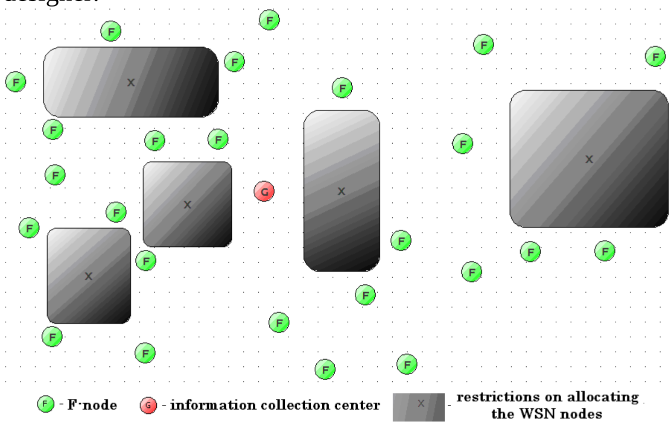

Figure 2. Example of the initial allocation of F-nodes and the information collection center

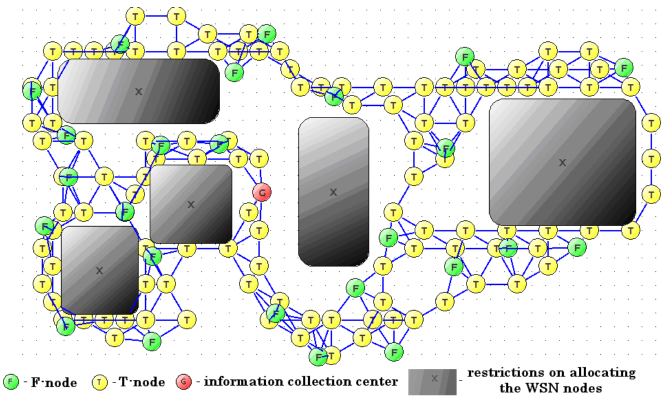

Figure 3. Example of the designed fault-tolerant WSN structure (every F-node has at least 4 independent paths to the ICC)

III. The Simplified Flow Chart of the Wireless Sensor Networks Construction Process

The process of designing a wireless sensor network (WSN) is rather complicated. This process is not formalized in the form of a hard set of rules, algorithms and standards that guarantee the construction of WSN satisfying different requirements of the designer.

Basing on the functional flow chart of the design process of fault-tolerant WSNs [2] and the research conducted in works ([3]-[5], etc.), we propose in Fig. 4 the simplified functional flow chart of the WSN construction process.

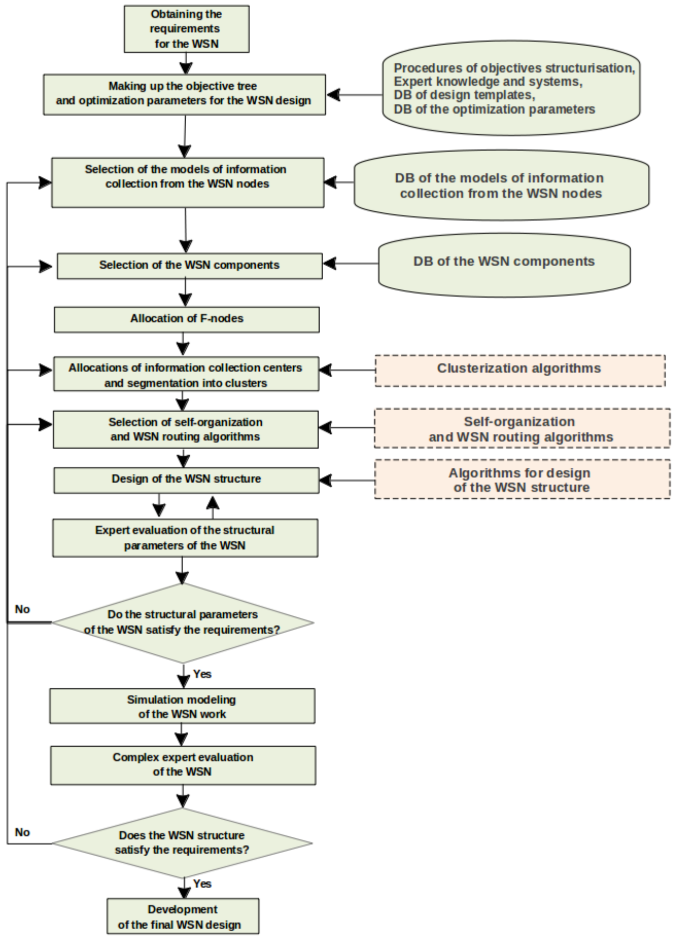

Figure 4. Simplified functional flow chart of the WSN construction process

The optimization parameters of the WSN structure design are formed in the leaf modes of objective tree (Fig. 4). Examples of optimization parameters may be: probability of connection between F-nodes and the ICC; coefficient of network readiness; viability parameters (for example, number and fraction of aborted or functioning connections, mathematical expectation and the average fraction of number of died or survived nodes after a virtual attack on the edges or nodes); the total time of network functioning before the moment of its fault; amount of power consumed by the network/nodes in a fixed time interval; time to deliver messages from F-nodes to the ICC; time to deliver messages from the ICC to network nodes in any time moment; time of the network self-recovery after nodes faults; confidence coefficient for the data collected by the network; redundancy coefficient of the transmitted data; coverage area and the density of T-nodes allocation; network protection criteria (for example, network/nodes vulnerability to attacks, average time to dispose of the vulnerability, number of network vulnerabilities, criticality of attacking actions and threats); monetary cost of the network; monetary expenses for network allocation and operation etc.

In Fig. 4 we have noted the functional block of the WSN structure design (FB-D-WSN). The main task of FB-D-WSN is allocation of T-nodes in such way that the designed network structure would satisfy the requirements and objectives of design, i.e. would have the «desired properties» of the designer.

The initial data for FB-D-WSN are the following: location and type of F-nodes (for example, Fig. 2); location and type of ICCs (for example, Fig. 2); description of the object that the WSN needs to be located at (its dimensions, scheme, spatial requirements for T-nodes allocation, barriers for electromagnetic waves expansion and characteristics of these barriers); types and characteristics of ready-to-use T-nodes; adopted self-organization and routing algorithms; information collection model; functional requirements; optimization parameters; fuzzy expert systems etc.

IV. Applying bio-inspired algorithms for WSN design

In recent years, the research area of *Natural Computing* is rapidly developing. It unites mathematical methods in which the principles of natural mechanisms of decision making are embedded [6]. Scientists have developed bio-inspired algorithms (BA) modeling animals behavior ([7], [8], etc.]) for solving various optimization problems that either do not have exact solution, or the solutions search space is vary large and complex constraints of the objective function are present, as well as NP-complete.

The described recommendations on applying BA and the proof in [9] that even the constrained variant of the problem of minimal coverage on plane is NP-complete allow to conclude about the possibility to apply self-organizing bio-inspired algorithms for a self-organizing WSN structure design.

Bio-inspired algorithms can be seen as multi-agent systems, each agent in which operates autonomously on very simple rules [10]. The most frequently used bio-inspired agents (B-agents) include: ants, bees, termites, fireflies, birds, fish, bats, cats, wolves, etc (Fig. 5).

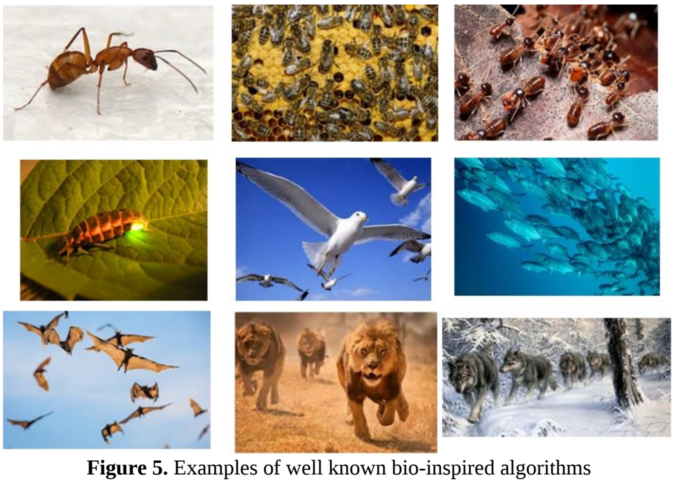

Figure 5. Examples of well known bio-inspired algorithms

In work [4] the author propose to jointly use modifications of ant and bee algorithms for the WSN structure design. In Fig. 6 we propose a generalized functional flow chart of a multi-agent bio-inspired design of the WSN structure, which is based on the use of B-agents of the shared global memory of stored pheromone (SGMSP). That is, SGMSP acts as a repository of knowledge (experience) for all B-agents. To use the following flow chart it is needed to define rules to move through the T-nodes for each type of B-agent and to take into account the use of SGMSP in these rules.

- **Step 1-3:** determine the model and the evaluation function for probability of presence of wireless channel with the desired properties in a heterogeneous space between two arbitrary points of the WSN allocation object in time T,
- **Step 2:** define a set M_{ALL} of all optimization parameters; the functions for calculating the parameters of M_{ALL}; a subset of optimization parameters $M_1 \subset M_{ALL}$,
- **Step 3:** determine the membership functions of fuzzy sets that characterize the optimization parameters of the M_{ALL}; fuzzy expert system shown in Fig. 6; fuzzy expert system to derive the confidence factor to meet the functional requirements of the designer,

Create an empty set of the best solutions Ω_{BEST} . Determine the maximum number of solutions b_K that will be stored in Ω_{BEST} .

Create a variable k for varying the density of T-nodes of the mesh which is initialized with the value 1 (k=1); compute the smallest distance D_{LOW} of the confident radio transmission among all nodes (F-nodes, T-nodes and ICC).

- **Step 4:** In RAM, form a mesh grid covering the object of the WSN. Set the length of the cell side to the value $D = \frac{D_{LOW}}{k}$. In each grid cell, place one T-node of each type. The mesh networks shown in Fig. 7 can be used. An equilateral triangle is the most appropriate cell for covering the mesh. Create an empty set Ω_T and add all of the T-nodes of the mesh to it.

The Nobel prizewinner in Physiology or Medicine Karl von Frisch analyzed the reasons for the bees to select hexagonal cell shape and noted that "with round, octa- or pentagonal cells the unused space remains; in addition, each cell would have its wholly or partly own walls, that is, would require extra building material. With tria-, tetra- or hexagonal cells both of these deficiencies disappear. However, among all three equal in area geometric shapes the hexagons have the smallest perimeter. Therefore, for the construction of hexagonal cells with the same capacity the smallest amount of building material is needed" [11].

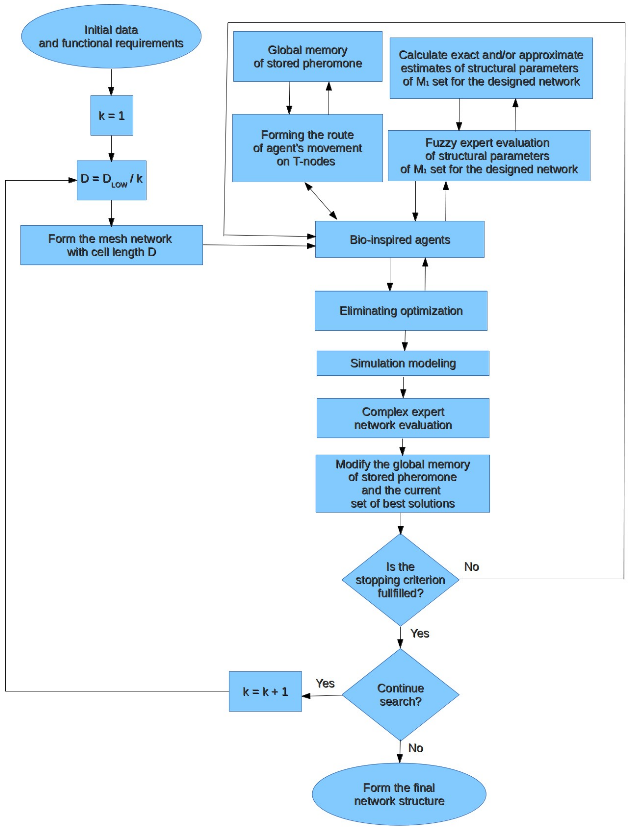

Figure 6. Functional flow chart of the multi-agent bio-inspired WSN structure design

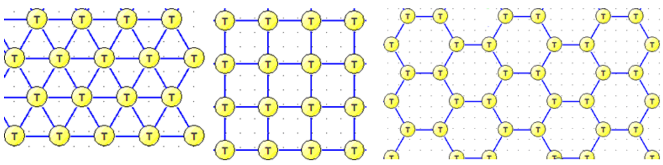

Figure 7. Ready-to-use mesh networks

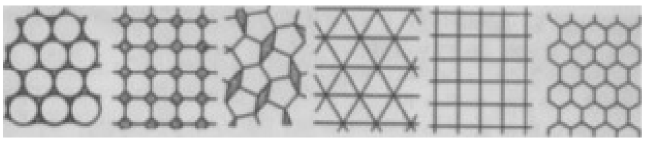

Figure 8. Why do the bees use hexagonal cells? [11]

Create a global memory of the stored pheromone. The pheromone is stored on the edges of a fully connected undirected weighted graph (FCUWG), the nodes of which are T-nodes. To store the edges of the graph in computer memory it is required to create a two-dimensional array *feromoneNetwork* with $\frac{N(N-1)}{2}$ memory cells of type *float*, where N is the number of T-nodes. The program code in *Java* language to allocate the memory for the storage of pheromone on FCUWG edges is shown below:

```
float feromoneNetwork[][] = new float[N-1][];
for(int i = 0; i < N; i++)
    feromoneNetwork[i] = new float[N - (i + 1)];
```

Initialize as a constant the value of the pheromone trace (the values of the feromoneNetwork array) on all edges of the graph.

- **Step 5:** Execute bio-inspired multi-agent algorithms. We suppose that there is a fitness function $C(n)$ of multi-criteria evaluation in the interval [0, 1] , defining quality of the T-node of any type. We suppose that there is a choice function $F_L(k)$ using to select the set Ω_{Lk} of favorite types of T-nodes for the next k-th agent.

A.1. Create a two-dimensional array *feromoneDif* to store changes in the pheromone using the following code:

```
float feromoneDif [][] = new float[N-1][];
for(int i = 0; i < N; i++)
feromoneDif [i] = new float[N - (i + 1)];
```

All values of *feromoneDif* must be initialized as zeros (in the above code, the zero-initialization is done automatically).

Define the number of different bio-inspired agents m, the strategy for choosing the initial location of the agent and other parameters needed for the agent to perform the work.

A.2. For each agent, perform the following steps (the code can be parallelized, i.e. to run in a separate thread for each agent):

A.2.1. Form, using the movement rules of the agent, the SGMSP, functions $C(n)$ and $F_L(k)$ the route M_T (array) of agent moving on T-nodes Ω_T ,

A.2.2. Create an empty extensible array of T-nodes M_{STR}, in which the T-nodes of the designed structure will be placed,

A.2.3. Select the design strategy:

(a) <u>sequentially add T-nodes to the network</u>. Go to step A.2.4;

(b) <u>sequentially remove T-nodes from the network</u>. Add to M_{STR} all nodes from MT array in the same sequence order. Go to step A.2.11;

A.2.4. Create an empty set H_P , which will contain the caches of such internal parameters of the functions of computing estimates M_1 , which will increase the speed of computing estimates M_1 for the next iteration. Create a variable i to store the index of the current T-node from the M_T array and initialize its value to 0 (i=0). Set the node T_C (T_C=M_T[0]) as the current one,

A.2.5. Add to M_{STR} the T-node T_C. Form the network structure S_S of T-nodes M_{STR},

A.2.6. Calculate using the caches H_P the values estimations of the optimization parameters of the set M_1 having structure

S_S . Clear H_P . Save the caches of the internal parameters of the functions of computing estimates M_1 of the current iteration to the set H_P ,

A.2.7. With use of a fuzzy expert system, calculate the reliability coefficient K_{D1} of meeting the requirements of the designer for parameters of a set M_1 of structure S_S ,

A.2.8. If $K_{D1} > p_1$, where p_1 is a set threshold, go to step A.2.11,

A.2.9. If $i<| M_T |$, put $i = i + 1$ and accept the next T-node $T_C = M_T[i]$ as the current one. Repeat steps A.2.5. – A.2.9. while i does not become equal to $| M_T |$,

A.2.10. Exit with notification of the failure from the function of agent design of the network structure,

A.2.11. Steps of eliminating optimization:

A.2.11.1. Select the strategy of eliminating optimization:
(a) step-by-step optimization with consideration of optimization parameters M_1. The fuzzy expert estimation of the structural parameters M_1 is used;
(b) step-by-step optimization with consideration of all optimization options M_{ALL}. The unit of simulation modeling and complex assessment of the network is used;
A.2.11.2. Revert the M_{STR} array,

A.2.11.3. In a loop, temporarily exclude each T-node $T_O \in M_{STR}$ from M_{STR}, then compute the confidence factor K_D of meeting the requirements of the parameters of eliminating optimization strategy. If in absence of T-node T_O evaluation of network structure stops meeting the designer requirements, put T_O back to M_{STR} into its place.

A.2.11.4. Revert the M_{STR} array,

A.2.12. Perform simulation modeling of the network. The results of the modeling and structural-parametric estimates of the various parameters are the input to the complex expert system for evaluation of network structure. Calculate with the latter the confidence factor K_{DALL} of meeting all the requirements of the designer,

A.2.13. If $K_{DALL} > 0$ then, in accordance with one of the following strategies, increase the pheromone amount in the array *feromoneDif*:

(a) consequent update – increase the amount of pheromone on the edges of the agent sequential traveling on T-nodes of M_{STR} by the value equal to $\Delta\tau_{ij,k}(t) = Q_{agent} \cdot K_{DALL}$, where Q_{agent} is the amount of pheromone secreted by the agent on one edge;
(b) full-mesh update – increase the amount of pheromone on all edges of the fully connected graph constructed on the basis of T-nodes of M_{STR} by the value equal to $\Delta\tau_{ij,k}(t) = Q_{agent} \cdot K_{DALL}$.

A.2.14. If K_{DALL} is greater that the estimate of the worst solution from Ω_{BEST} , or ($|\Omega_{BEST}| < b_K$ and $K_{DALL} > 0$), then add into Ω_{BEST} the current solution. By the solution we mean the couple (M_{STR}, K_{DALL}) . If $|\Omega_{BEST}| \geq b_K$ then leave in Ω_{BEST} only b_K best solutions.

A.3. After all agents have performed step **A.2**, update SGMSP (*feromoneNetwork* array) in accordance with the following well-known rule [7]: $\tau_{ij}(t+1) = (1-p)\cdot\tau_{ij}(t) + \Delta\tau_{ij}(t)$, where $\Delta\tau_{ij}(t)$ is the amount of pheromone on edge (i,j) in the array of pheromone changes *feromoneDif*, and $p \in [0,1]$ is the coefficient of pheromone evaporation. To enhance the intermediate best solutions, the amount of pheromone on the edges of the routes of the best solutions Ω_{BEST} should be increased (an example is using "elite" ants).

A.4. If the stopping criterion is not met, go to step **A.1.**

- **Step 6:** If it is necessary to continue the search, then increase *k (k = k + 1)* and go to Step **4**.

- **Step 7:** Return the best solution from Ω_{BEST} **.**

V. Applying the Heuristics "Flower Consistency" in a Consequent Ant Algorithm for Design of Heterogeneous Structure of the WSN

Karl von Frisch in his work [11] noted the critical importance of flower scent to be found by bees. So, for example, Karl von Frisch writes that "if you look closely at the bees collecting honey on a meadow, you may notice the striking fact: a bee flies in a hurry from a clover on a clover, paying no attention to the rest of the flowers; the other one at the same time flies from a thyme to a thyme, and the third one acts as if it is interested only in forget-me-nots. Biologists call such behavior "flower consistency". This applies of course only to individuals and not to the entire family; when one group of bees collects nectar from clovers, other worker bees from the same hive can choose the aims of their flights to the forget-me-nots, thymes or other flowers. Flower consistency is beneficial both for bees and plants. For bees this is because they are, standing by the certain flowers, meet the same working conditions which they have become accustomed to. One must see how long does a bee that had flown on a specific flower for the first time touch it with its proboscis, until it finds the hidden droplets of nectar, and how smartly does she reach the goal afterwards: only then one can understand what a saving of time does flower consistency provide. This is because the more often does one repeat specific action, the better it performs. But even more important is such behavior of bees for flowers, because they are dependent on rapid and successful pollination; it is clear that the pollen of clover, for example, would be totally unsuitable for thyme" [11].

Hereafter we formalize the steps of applying the bees "flower consistency" heuristics by ants (stage A.2 of the step 5 of the generalized algorithm for structure design of the WSN, given in Section 4).

M-A.2. For each ant agent, perform the following steps including the "flower consistency" heuristics in order to form a transmission route on T-nodes.

A. Define the set Ω_{Lk} for the ant agent, using $F_L(k)$,

B. Create the set $J_{i,k}$ of T-nodes and add all T-nodes from Ω_T to it,

C. Create an empty set L_k of T-nodes and add all located in a mesh network T-nodes having types of Ω_{Lk} to it. Applying the "flower consistency" heuristics for ants movement consists in that initially the ant visits T-nodes of the set L_k.

D. Create an empty extensible array of T-nodes M_{STR}, in which the T-nodes of the designed structure will be placed,

E. Create an empty set H_P, which will contain the caches of such internal parameters of the functions of computing estimates M_1, which will increase the speed of computing estimates M_1 for the next iteration,

F. Select the initial location of ant (starting T-node T_C) and add T_C to M_{STR}. Remove T-node T_C from $J_{i,k}$.

G. If the ant has walked on all T-nodes having type of Ω_{Lk} (i.e., $|J_{i,k} \cap L_k| == 0$), then $L_k = J_{i,k}$ (i.e. the ant will further walk on T-nodes of "non-favorite" types),

H. Form the WSN structure S_S from T-nodes M_{STR}. Calculate using caches H_P the estimates of the values of optimization parameters of the set M_1 of S_S structure. Clear H_P. Save the caches of internal parameters of the estimates calculation functions M_1 of the current iteration in H_P set,

I. Using a fuzzy expert system, calculate the confidence factor K_{D1} of meeting the requirements of the designer for parameters of a set M_1 of structure S_S,

J. If K_{D1} is greater than threshold p_1, then go to step ***L***.

if $K_{D1} \le p_1$ and $|J_{i,k}| == 0$, then exit the function of constructing the ant's route on T-nodes with notice of failure,

if $K_{D1} \le p_1$ and $|J_{i,k}| > 0$, then, using the following modification of a known probabilistic-proportional rule [7] of ant movement, define the next T-node T_N of the ant's route.

$$\begin{cases} P_{ij,k}(t) = \dfrac{[\tau_{ij}(t)]^{\alpha} \cdot [\eta_{ij}]^{\beta} \cdot c_j^{\gamma}}{\sum\limits_{l \in J_{i,k} \cap L_k} [\tau_{il}(t)]^{\alpha} \cdot [\eta_{il}]^{\beta} \cdot c_l^{\gamma}}, \text{ если } j \in J_{i,k} \cap L_k, \\ P_{ij,k}(t) = 0, \text{ если } j \notin J_{i,k} \cap L_k \end{cases}$$

, where $P_{ij,k}(t)$ is probability of the *k*-th ant moving from node *i* to node *j* on the *t*-th iteration; $\tau_{ij}(t)$ is the pheromone trace on the edge (*i,j*) in the global memory of the stored pheromone (*feromoneNetwork* array); η_{ij} is the ant's visibility; c_j is the estimate of T-node *j* quality; α, β and γ are parameters defining weights of the pheromone trace, of visibility when selecting the route and T-node quality; $J_{i,k}$ is the set of nodes that the ant *k* being in node *i* still has to visit.

In addition to the strategies of calculating ant's visibility η_{ij} proposed in [4] ("keep away from ICC", "keep away from the node", "keep close to ICC", "keep close to the node", "absent") we add one more possible strategy "neighbour nodes", i.e.

$$\begin{cases} \eta_{ij} = 1, \text{ if node } j \text{ is adjacent to at least one node from } M_{STR}, \\ \eta_{ij} = 0, \text{ otherwise} \end{cases}$$

K. Add T_N to M_{STR}. Remove T-node T_N from $J_{i,k}$. Go to step G,

L. Execute the steps of eliminating optimization **A.2.11**.

M. Execute steps **A.2.12-A.2.14**.

VI. Eliminating Ant-Bee Algorithm of the WSN Structure Design

Michelangelo told that there is a statue in every stone and all that is needed is to be able to remove all the excess and to elicit it (Fig. 9). Having the initial "stone" formed (on step 4 of the generalized algorithm of the WSN structure design in Section 4), we assign a unique integer index to each T-node and perform search of a sequence of removing excessive T-nodes in order to get desired WSN structure.

We perform the hybride and-bee algorithm for finding a sequence of removing excessive T-nodes in order to get desired WSN structure. Let us consider realization of ant and bee algorithm.

Figure 9. There is a statue in every stone, and all that is needed is to be able to remove all the excess and to elicit it.

6.1. Application of ant algorithm

We use the known probabilistic-proportional rule of ants movement [7]:

$$\begin{cases} P_{ij,k}(t) = \dfrac{[\tau_{ij}(t)]^{\alpha} \cdot [\eta_{ij}]^{\beta}}{\sum\limits_{l \in J_{i,k}} [\tau_{il}(t)]^{\alpha} \cdot [\eta_{il}]^{\beta}}, \text{ если } j \in J_{i,k}, \\ P_{ij,k}(t) = 0, \text{ если } j \notin J_{i,k} \end{cases},$$

In the work we present the following strategies of ant's visibility calculation in the problem of interest:

1. "Keep away from the ICC" - the further is T-node *j* from the ICC, the better is it visible to the ant being at the node *i* (i.e. $\eta_{ij} = \dfrac{D_{j\_BS}}{MAX_{T\_BS}}$, where D_{j_BS} is the distance between T-node *j* and the ICC, MAX_{T_BS} is the distance from the outermost T-node to the ICC;

2. "Keep away from the node" - the further is the ant from T-node *j*, the better is it visible to the ant being at the node *I* (i.e. $\eta_{ij} = \dfrac{D_{i\_j}}{MAX_{T\_i}}$, where D_{i_j} is the distance between T-nodes *i* and *j*, MAX_{T_i} is the distance from T-node *i* to the outermost T-node from it;

3. "Keep close to the ICC" - the closer is T-node *j* to the ICC, the better is it visible to the ant being at the node *i* (i.e. $\eta_{ij} = \dfrac{MAX_{T\_BS} - D_{j\_BS}}{MAX_{T\_BS}}$);

4. "Keep close to the node" - the closer is the ant to T-node *j*, the better is T-node visible to the ant being at the node *i* (i.e. $\eta_{ij} = \dfrac{MAX_{T\_i} - D_{i\_j}}{MAX_{T\_i}}$);

5. "None", i.e. $\eta_{ij} = 1$.

To select the initial location of the ant colony one can use the following strategies [10]: "Shotgun", "Blanket", "Wandering colony" and "Focusing".

Each ant agent works with a copy of the WSN structure which will be changed upon performing the function of assessing the fitness of the route (sequence of removing the T-nodes).

To evaluate the *fitness function* of the route which the ant has passed on, one should perform the following algorithm:

1. Create an empty set of T-nodes M_{DEL} which will contain the removed T-nodes;
2. Create an array of T-nodes M_T sorted according to the route of ant's movement; form the WSN structure from T-nodes of M_T.
3. In a loop, temporarily remove each T-node $T_C \in M_T$ from the WSN structure, afterwards perform fuzzy expert evaluation of the structural parameters $M_1 \subset M_{ALL}$. If in absence of the temporarily removed T-node T_C the evaluation of the WSN structure meets designer's requirements then add T-node T_C to M_{DEL} and remove T_C from the WSN structure forever. Otherwise, return T_C into the WSN structure. After having checked the possibility to remove every T-node, perform simulation modeling of the WSN work and calculate the final confidence factor K_{DALL} of meeting all designer's requirements.

4. As a result of performing step 3, $r = |M_{DEL}|$ T-nodes have been removed from the WSN structure. According to the logic of step 3, any path of ant's movement which starts from M_{DEL} T-nodes will return the same WSN structure as a result of executing the function of evaluating the route fitness. In classical works on applying ant algorithms to solve traveling salesman problem, the pheromone is updated on the edges of the found path. In this work we use the "Factorial" strategy of increasing the pheromone amount on those edges of the graph that connect $\dfrac{r(r-1)}{2}$ unique pairs of remote T-nodes for a value equal to $\Delta\tau_{ij,k}(t) = K_{DALL} \cdot Q$, where $\Delta\tau_{ij,k}(t)$ is an amount of pheromone secreted by the ant *k* on the edge (*i*,*j*), *Q* is an amount of pheromone secreted by the ant on one edge. In such a way the ant agent increases the amount of pheromone on *r*! possible routes.

After all ants of the colony have performed evaluation of the found routes fitness, the pheromone must be updated.

Update the current set of best solutions Ω_{BEST}. We treat as a solution a couple (Ω_S, K_{DALL}), where Ω_S is a set of T-nodes left after removal ($\Omega_S = \Omega_T \setminus M_{DEL}$).

6.2. Application of bee colony algorithms in the problem of interest

Various models and algorithms of bee colony behavior are given in work [8]. In most models of a bee colony behavior there are the following concepts: scout bees, foraging bees (worker bees) and the nectar. Scout bees form the set of the promising areas of the search for solutions. Foragers explore the promising regions found by the scout bees. The purpose of the bee colony is to find sources that contain as much nectar as possible. The found amount of nectar represents the value of the objective function.

In [12] the following rules for foragers movement used for solving the travelling salesman problem are proposed:

$$P_{ij,n} = \frac{[\rho_{ij,n}]^{\alpha}[\frac{1}{d_{ij}}]^{\beta}}{\sum_{j \in A_{i,n}} ([\rho_{ij,n}]^{\alpha}[\frac{1}{d_{ij}}]^{\beta})},$$

where

$$\rho_{ij,n} = \begin{cases} \lambda & , j \in F_{i,n}, |A_{i,n}| > 1 \\ \frac{1-\lambda|A_{i,n} \cap F_{i,n}|}{|A_{i,n} - F_{i,n}|} & , j \notin F_{i,n}, |A_{i,n}| > 1 \\ 1 & , |A_{i,n}| = 1 \end{cases},$$

$P_{ij,n}$ is the probability of a bee to move from the node *i* to the node *j* on *n*-th transition, λ is the probability of a bee to move to a preferred node, $A_{i,n}$ is the set of available non-visited nodes on *n*-th transition, $F_{i,n}$ is the set of a single node defining the bee's preference on movement on the n-th transition from the node *i*, α and β are adjusted weight parameters.

In this work we propose a modification of given above rule of bees movement, which allows the bees to use for $P_{ij,n}$ calculation not only the preferred movement route, but also the current global memory of ants (pheromone), i.e. $P_{ij,n} = \frac{[\rho_{ij,n}]^{\alpha}[\tau_{ij}]^{\beta}}{\sum_{j \in A_{i,n}} ([\rho_{ij,n}]^{\alpha}[\tau_{ij}]^{\beta})}$, where τ_{ij} is the pheromone trace on the edge (*i*,*j*). In Fig. 10 there is an analogy of bees "overseeing" ants movement routes, as in the proposed modification.

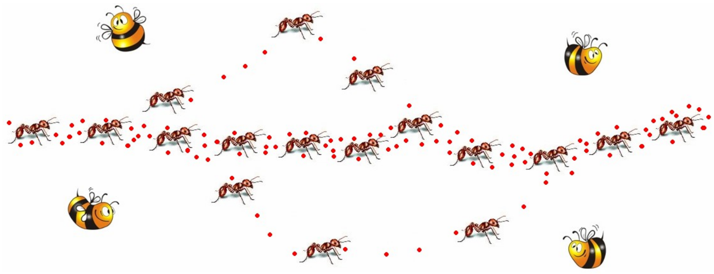

Figure 10. Analogy of bees "overseeing" ants movement.

For every worker bee one should choose the flight starting point (a place to install the hive) and the preferred flight route which is formed basing on one of the best solutions of the set Ω_{BEST}. For choosing one solution from Ω_{BEST}, one should adapt either the considered in [12] probabilistic rules of choosing a perspective solution (based on the "waggle dance" performed by bees) or the selection method for genetic algorithms based on a roulette wheel principle [13] (i.e. the probability of selecting one solution from Ω_{BEST} is proportional to reliability coefficient K_D for this solution).

After one of the best solutions have been chosen (Ω_S, K_{DALL}), set the bee's flight starting point into a random T-node from the set $M_{DEL} = \Omega_T \setminus \Omega_S$ and form the preferred rout of bee's flight in the following way: 1. create an array of T-nodes M_F to store the preferred sequence of T-nodes of the bee's flight; 2. add to M_F randomly shuffled elements of the set $M_{DEL} = \Omega_T \setminus \Omega_S$; 3. add to M_F randomly shuffled elements of the set Ω_S.

After the bee has flown around all nodes, one should execute the considered in Section 6.1. *function of route fitness evaluation*, and if in a current set of best solutions there are ones worse that the one found by the bee, then update the current set of best solutions.

In order to save the result of bee's flight in the global memory of the stored pheromone (that provides "mutually beneficial symbiosis" between bees and ants), implement, in accordance with "Factorial" strategy, increment of the pheromone amount on the edges of the graph connecting $\frac{r(r-1)}{2}$ of unique pairs of remote T-nodes by the value equal to $\Delta\tau_{ij} = K_{DALL} \cdot Q_{BEE}$, where Q_{BEE} is the amount of pheromone secreted by the bee.

6.3. Shifting to the next iteration in combined application of ant and bee algorithm

In order to reduce the number of T-nodes in the current coverage (reducing the search space of solutions), on each *t*-th iteration of ant-bee algorithm work one should remove from the coverage those T-nodes that do not belong to any set Ω_S of the best solutions and the solutions resulting from use of ant and bee algorithms in the last *t* iterations.

6.4. Results of work of the software-implemented ant-bee algorithm of the WSN structure design

The ant-bee algorithm for the WSN structure design software is implemented as a software module for the system [14]. The results of the algorithm are shown in Figure 3. In this example we need to allocate the T-nodes in such way that upon failure of any *n*=3 T-nodes, each F-node will be able to transfer data to *K*=1 ICC while minimizing the cost (total number of allocated T-nodes). Figure 11 shows the evolution of the WSN structure in the process of finding a rational solution at different iterations of the algorithm work. Figure 12 shows the reduction of the best solution search space at each *t*-th iteration, where *t*=8.

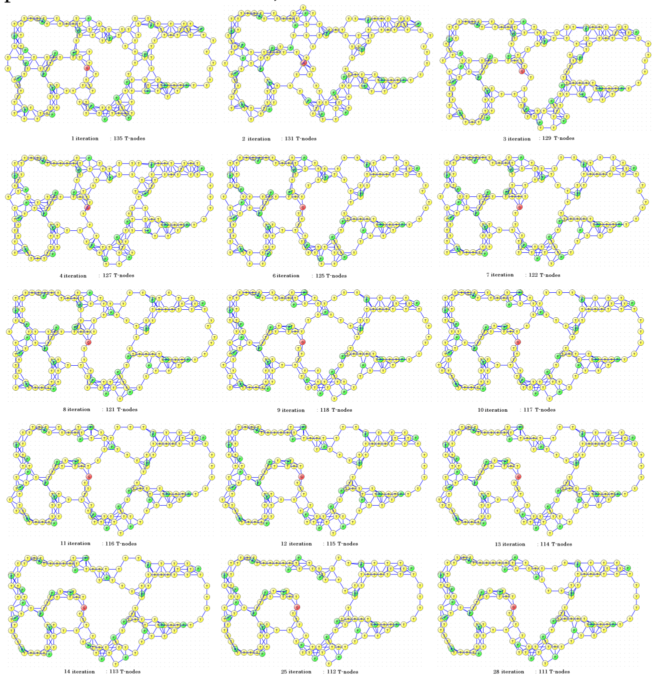

Figure 11. Evolution of the WSN structure at different iterations of algorithm work.

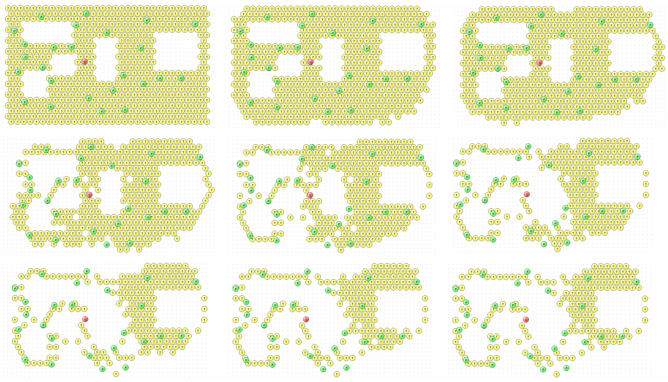

Figure 12. Reducing the best solution search space.

VII. CONCLUSIONS

The bioinspired multi-agent algorithms proposed in this work can be used to find approximate solutions to the problem of constructing a wireless sensor network taking into account spatial constraints on the placement of network nodes.

The algorithms were implemented programmatically in *Java* and became the basis of the system of wireless sensor networks design.

Experimental research of proposed algorithms has shown the possibility of constructing the WSN structure considering various objective functions and optimization parameters (given in e.g. Section 3).

The considered algorithms for constructing the WSN structure are compared with a known algorithm NTRRP+optimization [15]. In the test task it was required to allocate transit nodes in such way that each F-node would have at least 3 independent paths to the ICC. The results of the algorithms are shown in Fig. 13-15. The algorithm NTRRP with subsequent optimization has allocated 147 T-nodes (Figure 13).

The ant algorithm has allocated 102 T-nodes (Figure 14) with k=1 and using strategies: constructing WSN structure “sequentially remove T-nodes from the network”; updating the pheromone “full-mesh update”; eliminating strategy “step-by-step optimization with consideration of optimization parameters M_1”; ant visibility strategy “neighbour nodes”.

Eliminating ant-bee algorithm of the WSN structure design allocated 99 T-nodes (Figure 15) with k=2.

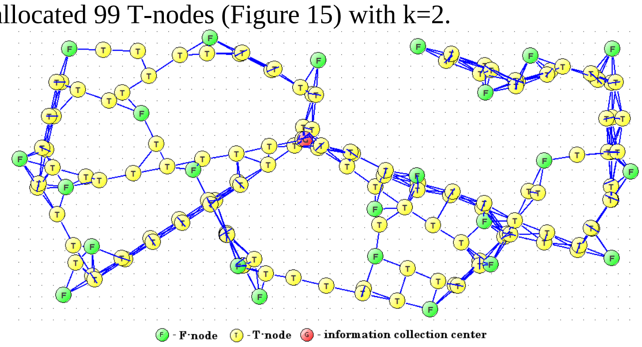

Figure 13. NTRRP with optimization [15] allocated 147 T-nodes.

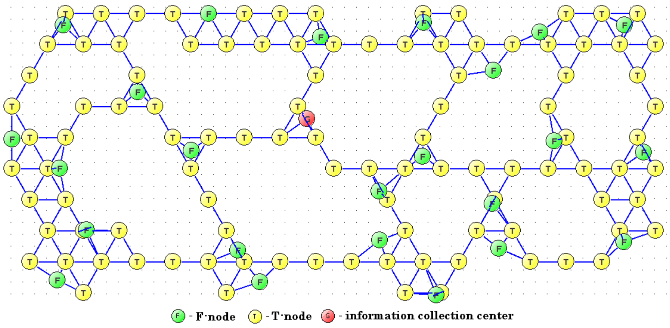

Figure 14. The ant algorithm with “sequentially remove T-nodes from the network” strategy allocated 102 T-nodes with k=1.

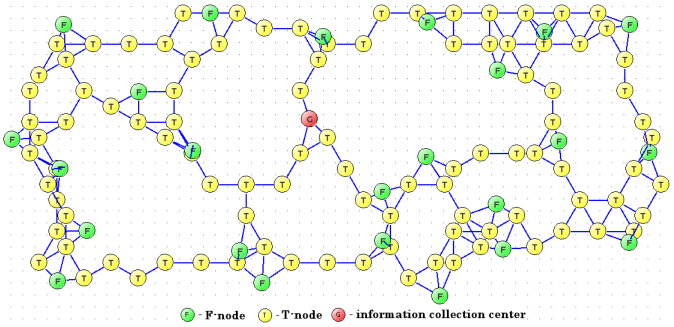

Figure 15. Eliminating ant-bee algorithm of the WSN structure design allocated 99 T-nodes with k=2.

ACKNOWLEDGMENT

The reported study was supported by RFBR, research project No15 07-09431A “Development of the principles of construction and methods of self-organization for Flying Ubiquitous Sensor Networks”

REFERENCES

[1] R. Faludi. “Building Wireless Sensor Networks”, O'Reilly Media, 2010, 320 p.
[2] V.A. Mochalov, E.N. Turuta. “Functional diagram of the wireless monitoring networks design process”, Sensors and Systems, vol.2, pp. 40-44, 2010.
[3] V.A. Mochalov. “The method of fault-tolerant sensor network structure synthesis with restrictions on placement of nodes in heterogeneous space”, T-Comm, vol.10, pp. 71-75, 2012.
[4] V.A. Mochalov. “Hybrid bionic algorithm for wireless sensor network structure synthesis” , T-Comm, vol.10, pp. 72-77, 2013.
[5] V.A. Mochalov. “Algorithms for increase sensor network total working time to the moment of its failure”, Devices and systems . Management, control , diagnostics, vol.7, pp. 12-19, 2010.
[6] G. Rozenberg, T. Bäck, J.N. Kok. “Handbook of Natural Computing”, Springer, ISBN 978-3-540-92909-3, 2012.
[7] F. Bonavear, M. Dorigo, G. Theraulaz. “Swarm Intelligence: from Natural to Artificial Systems”, New York, Oxford University Press, 320 p, 1999.
[8] D. Teodorovic. “Bee Colony Optimization (BCO)”. Available: http://www.sf.bg.ac.rs/downloads/katedre/oi/1.BCO-Book-Chapter.pdf .
[9] R.J. Fowler. “Optimal packing and covering in the plane are NP-complete”, Inf. Process. Lett., vol.12(3), pp.133–137, 1981.
[10] A.A. Kazarov, V.M. Kureichik. “Bioinspired algorithms. Solution of optimization problems”, LAP LAMBERT Academic Publishing, 80 p, 2011.
[11] K.V. Frisch. “Bees, their vision, chemical senses, and language”, Ithaca, N.Y.: Cornell University Press, 152 p.,1971.
[12] L. Wong, M. Y. H. Low, C. Chong. “An Efficient Bee Colony Optimization Algorithm for Traveling Salesman Problem using Frequency-based Pruning”, Industrial Informatics, 7th IEEE International Conference on 23-26 June 2009, pp. 775 – 782.
[13] B. Thomas. “Evolutionary Algorithms in Theory and Practice”, Oxford Univ. Press, 120 p., 1996.
[14] V.Mochalov. Certificate on the state registration of the computer program “Intellectual support system of Fault-Tolerant sensor networks design”, № 2009615483 from 02.10.2009, Russia.
[15] M. Ahlberg, V. Vlassov, Y. Terumasa. “Router Placement in Wireless Sensor Networks”, IEEE International Conference on Mobile Adhoc and Sensor Systems (MASS), pp. 538–541, 2006.

Vladimir Mochalov was born in Lyubertsy, Russia in 1985. He received the Ph.D. degree in electronic engineering from Moscow Technical University of Communications and Informatics. His research interests include networks structure synthesis, artificial intelligence, bio-inspired algorithms, query answering systems and Big Data.